PACIFIC NANOTECHNOLOGY

PScan2™ SPM Controller

©2002 Pacific Nanotechnology, Inc.
3350 Scott Boulevard • Building #29
Phone 408.982.9492 • Fax 408.982.9151

Table of Contents

Important Information i

Product Warranty ii-iv

Software License v

Copyright Information vi

Safety Statement vii

CHAPTER 1 - Indtroduction

1.1 About the PScan2™Controller 1-2

1.2 What You Need to Get Started 3

1.3 Software Programming Choices 4

1.4 Optional Equipment 5

1.5 Unpacking 6

CHAPTER 2 - Installation & Ethernet Configuration

2.1 Installation 7

2.2 Hardware Configuration 8-9

CHAPTER 3 - Description of External Signal Connections

3.1 I/O Connector Pin Description 10-12

CHAPTER 4 - Description of Operation

4.1 Overview 13-14

4.2 Detailed Block Diagram 15

4.3 Description 16-17

CHAPTER 5 - DCEx™

5.1 PScan2™ System Configuration 18-21

5.2 Commands & Controller Functional Modes 22-38

CHAPTER 6 - SPMCockpit™ User Interface

6.1 Introduction 39

6.2 Description of Contents 39-41

APPENDIX A 42-51

APPENDIX B 52-62

APPENDIX C 63

APPENDIX D 64-65

APPENDIX E 66-68

APPENDIX F 69-88

APPENDIX G 89-104

APPENDIX H 105-108

APPENDIX I 109

APPENDIX J 110-124

APPENDIX K 125-127

Important Information

How to Contact Us

Technical Support
Pacific Nanotechnology, Inc.
3350 Scott Blvd #29
Santa Clara, CA 95054-3105

Telephone Support (U.S.)
Telephone: (408) 982-9492
Fax: (408) 982-9151
Web Address: http:// www.pacificnanotech.com
E-mail:info@pacificnanotech.com

PACIFIC NANOTECHNOLOGY PRODUCT WARRANTY

Coverage

Pacific Nanotechnology warrants that products manufactured by Pacific Nanotechnology will be free of defects in materials and workmanship for one year from the date of shipment. The product warranty provides for all parts (excluding consumables and maintenance items), labor, and software upgrades.

Instruments, parts, and accessories not manufactured by Pacific Nanotechnology may be warranted by Pacific Nanotechnology for the specific items and periods expressed in writing on published price lists or quotes. However, all such warranties extended by Pacific Nanotechnology for those specific items and periods expressed in writing on published price lists or quotes are limited in accordance with all the conditions, terms and other requirements noted in this warranty. Pacific Nanotechnology makes no warranty whatsoever concerning products or accessories not of its manufacture except as noted.

Customers outside the United States and Canada should contact their local Pacific Nanotechnology representative for warranty information that applies to their locales.

CUSTOMER RESPONSIBILITIES

- Complete ordinary maintenance and adjustments as stated in Pacific Nanotechnology manuals.
- Use only Pacific Nanotechnology replacement parts.
- Use only Pacific Nanotechnology approved consumables such as filters, lamps, cantilevers, etc.
- Provide safe and adequate working space for servicing of the products by Pacific Nanotechnology personnel.

REPLACEMENTS AND REPAIRS

- Any product, part, or assembly returned to Pacific Nanotechnology for examination or repair must have prior approval.
- A Return Materials Authorization or RMA number obtained from Pacific Nanotechnology prior to shipment must identify a return.
- It must be returned freight prepaid to the designated address by the customer.
- Return freight costs will be prepaid by Pacific Nanotechnology if the product, part or assembly is defective and under warranty.
- Pacific Nanotechnology will either replace or repair defective instruments or parts at its option.
- Repair and replacement of instruments or parts does not extend the time of the original warranty.
- Replacement parts or products used on instruments out of warranty are themselves warranted free of defects in materials and workmanship for 90 days with the exception of consumables such as filters, lamps, cantilevers, etc.

WARRANTY LIMITATIONS

This warranty does not cover:

1. Any loss, damage, and or product malfunction caused by shipping or storage, accident, abuse alteration, misuse, or use of user-supplied software, hardware, replacement parts, or consumables other than those specified by Pacific Nanotechnology.
2. Parts and accessories that are expendable and replaceable in the course of normal operation.
3. Products not properly placed and installed per our installation instructions.
4. Products not operated within the acceptable parameters noted per our installation instructions.
5. Products that have been altered or customized without prior written authorization from Pacific Nanotechnology.
6. Products that have had their serial number removed, altered or otherwise defaced.
7. Improper or inadequate care, maintenance, adjustment, alteration, or calibration by the user

Software License

Source Code License Agreement by:
Pacific Nanotechnology, Inc

You, the Licensee, assume responsibility for the selection of the program to achieve your intended results, and for the installation, use, and results obtained from the program.

IF YOU USE, COPY, MODIFY, OR TRANSFER THE PROGRAM, OR ANY COPY, MODIFICATION, OR MERGED PORTION, IN WHOLE OR PART, EXCEPT AS EXPRESSLY PROVIDED FOR IN THIS LICENSE, YOUR LICENSE IS AUTOMATICALLY TERMINATED.

LICENSE

You may:
Use the program on a single machine and copy the program into any machine-readable or printed form for backup or support of your use of the program on the single machine.

Modify the program and/or merge it into another program for your use on the single machine. Any portion of the program merged into another program will continue to be subject to the terms of this Agreement. You must reproduce and include the copyright notice on any copy, modification, or portion merged into another program.

Transfer the program and license to another party if either party agrees to accept the terms and conditions of this Agreement. If you transfer the program, you must at the same time either transfer all copies, whether in machine readable form or printed form, to the same party or destroy any copies not transferred; this includes all modifications and portions of the program merged into other programs.

TERM
The license is effective on the date you accept this agreement, and remains in effect until terminated as indicated above or until you terminate it. If the license is terminated for any reason, you agree to destroy the program together with all copies, modifications, and merged portions in any form.

Copyright Notice – covers all attached documents
©Pacific Nanotechnology Incorporated 2001-2002. All rights reserved.

Pacific Nanotechnology retains all ownership rights to this documentation and all revisions of the PScan2™Controller computer program and other related software options.

Reproduction of any portion of this document or any image depicted in this publication without prior written authorization (with the exception of archival purposes or for the specific use of the program with Pacific Nanotechnology equipment) is prohibited by law and is a punishable violation of the law.

PACIFIC NANOTECHNOLOGY INCORPORATED PROVIDES THIS PUBLICATION "ASIS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OR CONDITIONS OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO WAY SHALL PACIFIC NANOTECHNOLOGY INCORPORATED BE LIABLE FOR ANY LOSS OF PROFITS, LOSS OF BUSINESS, INTERRUPTION OF BUSINESS, LOSS OF DATA, LOSS OF USE, OR FOR SPECIAL, INCIDENTAL, INDIRECT, OR CONSEQUENTIAL DAMAGES OF ANY KIND EVEN IN THE EVENT OF SUCH DAMAGES ARISING FROM ANY DEFECT OR ERROR IN THIS PUBLICATIONS OR IN THE X'PERT MODE™ OR EZ MODE™ SOFTWARE.

The trademarks or registered trademarks of Pacific Nanotechnology are PScan2™, Nano-R™, X'Pert™ Mode and EZMode™

Safety Statement

WARNING REGARDING MEDICAL AND CLINICAL USE OF PACIFIC NANOTECHNOLOGY, INC. PRODUCTS

Pacific Nanotechnology, Inc. products are not designed with approved components and testing procedures intended to ensure a level of reliability suitable for use in treatment and diagnosis of humans. Applications of Pacific Nanotechnology, Inc. products involving medical or clinical treatment can create a potential for accidental injury caused by product failure, or by errors on the part of the user or application designer. Any use or application of Pacific Nanotechnology, Inc. products for or involving medical or clinical treatment must be performed by properly trained and qualified medical personnel, and all traditional medical safeguards, equipment, and procedures that are appropriate in the particular situation to prevent serious injury or death should always continue to be used when Pacific Nanotechnology, Inc. products are involved. Pacific Nanotechnology, Inc. products are NOT intended to be a substitute for any form of established process, procedure, or equipment used to monitor or safeguard human health and safety in medical or clinical treatment.

Chapter 1

Introduction

This chapter describes the PScan2™ Controller, lists what you need to get started, describes software programming choices, optional equipment, and custom cables, and explains how to unpack the PScan2™ Controller.

1.1 About the PScan2™ Controller

Thank you for purchasing the Pacific Nanotechnology, Inc. PScan2™ Controller.

The PScan2™ multiple SPM Controller sets a new performance standard: the operation of multiple scanning microscopes under the control of one master workstation. This new standard allows the user to operate one or more slave controllers that are linked to one main computer. Therefore, one overlay program within the master work station can control the scanning parameters and display acquired image data obtained from each slave controller. No other company offers such a multiple-unit SPM controller system.

The PScan2™ Controller concept represents one of the most cost-effective advances for controlling one or more Scanning Probe Microscopes. Each PScan2™ Controller contains a complete high speed PC computer containing data I/O boards which provide scan control parameter settings, data acquisition and image storage capability for each individual Scanner. Each PScan2™ Controller is connected to a Master Workstation via an Ethernet link, as indicated in Figure 1-1. Image data is acquired independently from each scanner/controller and transferred to the Master Workstation on request. The Master Workstation, operated under Windows 95 ™and Windows NT ™ environment, can then coordinate other tasks which will facilitate specimen throughput, such as automated specimen handling.

Slave Controller

In Appendix C-1, a simplified block diagram is shown for the Slave PC and Interface Board that comprise the PScan2™ Controller. Note that the PScan2™ Controller is designed to drive the stacked-type piezo drivers, such as those that are incorporated in TopoMetrix's large-range scanners. The detection and feedback circuitry incorporate a number of features that are typically found in more costly controllers. These include: modulation and demodulation circuits for oscillating cantilever modes, analog feedback linearizers for X and Y piezo drivers, conditioning (filter, gain and offset) for various data signals, and on-board stepper motor drivers. Other functions are outlined in the attached Specification Summary.

Software

The PScan2™ Controller communicates with the Master Workstation via an Ethernet file-structure communication software. The slave controller operates under DOS 6.22™ operating environment while the Master Workstation operated under Windows 95™ or Windows NT™. By using Microsoft Visual Basic™ programming, the user will have access to proprietary DOS-based drivers located on the slave for handling all the essential scan control functions. It is important to note that there is an Open Architecture Access for both the hardware as well as the Visual Basic™ level software.

1.2 What You Need to Get Started

To set up and use your PScan2™ Controller, you will need the following:

- PScan2™ Controller
- PScan2™ User Manual (this manual)
- Scanning Head or other input device
- Interface Cabling between Scanning Head and PScan2™ Controller
- The following software packages and documentation:
  - DCEx™ Protocol Driver Software
- Visual Basic Utility
- Your Master computer with Windows 95, 98, NT, or Windows XP™

Master Workstation, minimum requirements:

- Pentium 133 MHZ or faster processor
- 32 MB RAM memory
- 1 GB Hard Disk (larger preferred, as needed by user)
- 1 GB removable media Hard Drive (optional, as needed by user)
- 3.5 inch Floppy Drive
- 100 Mbit/sec Ethernet interface
- Video capability: as needed by user

1.3 Software Programming Choices

Visual Basic Application Software

There are several options to choose from when programming your Pacific Nanotechnology, Inc.. The ASPM-Cockpit™, which is included with the purchase of the PScan2™ controller, is a test utility and basic data acquisition program. It has been created in Microsoft Visual Basic™ language, a language that is convenient for the non-programmer to built and modify existing programs but includes extensive functions and flexibility with defined parameters.

Programming may also be performed in C++, a more comprehensive language and geared for meeting specific and more ranging requirements of the user.

PScan2™ Driver Software

The DOS 6.22™ code for controlling the PScan2™ hardware is proprietary to Pacific Nanotechnology, Inc.. The function calls and documentation for accessing these calls through Visual Basic ASPM-Cockpit™ program are available in this manual (see Chapter 5 and 6). At present, most, but not all, of the functions and capabilities of the controller can be accessed with the current version of the SPMCockpit™. New functions will be made available on a quarterly basis.

Should the user have specific needs, such as a new specific function or unique combination of functions for which the Controller software must be modified, please contact a Pacific Nanotechnology representative. An added function may be useful to other users.

Register-Level Programming

Under some circumstances, the user may need to access certain operations at the register level. For example, a complex series of high-speed I/O operations on the optional 16-bit data bus may be necessary. If this is the case, please contact a Pacific Nanotechnology representative.

1.4 Optional Equipment

Pacific Nanotechnology, Inc. offers a variety of products to use with your PScan2™ Controller, including cables, connector blocks, and other accessories, as follows:

- Cables and cable assemblies, no connector (open-ended) on user-side
- Scanner: 37-pin sub-D connector with 4-ft. cable; analog signals shielded
- Linearizer: 9-pin sub-D connector with 4-ft cable; analog signals shielded
- Signal Access: 50 pin dual-in-line with 6 ft. flat ribbon cable
- Steppers/Digital I/O: 60 pin dual-in-line with 6 ft. flat ribbon cable
- Signal Access Console
- Breakout box with BNC connectors for connecting to and monitoring various internal signals, external analog input and output signals and digital flags; includes 3 ft flat cable with 50-pin connectors on both ends.
- HV-450 Board
- For operating external quadrant tube-type scanners (+X,-X,+Y,-Y, Z)
- Including 5 high-voltage amplifiers with two power supplies (+/- 225 VDC)
- Factory Installed, includes internal cables

For pricing and more information about these products, please call our office.

Custom Cabling

Pacific Nanotechnology, Inc. offers cables and accessories for you to prototype your application or to use if you frequently change board interconnections.

You can interface the PScan2™ Controller to a wide range of scanner heads, test instruments, I/O racks and modules, screw terminal panels, and almost any device with a parallel interface. Please read through the detailed specification sheet and accompanying wiring diagrams to familiarize yourself with your options.

1.5 Unpacking

Your PScan2™ Controller is shipped in high impact packaging in order to prevent and/or minimize damage by mishandling. Please inspect your Controller box for external damage. If necessary, remove the outer 3-sided cover to visually inspect for internal damage. DO NOT POWER-UP! If there is reason to believe that the Controller was dropped, open the top cover to confirm that boards and cables appear to be seated. The top cover may be lifted up by removing twelve screws (torx head) from the front and rear panels (top and sides only) and four screws along each side, near the bottom, of the top cover.

Should there appear to be damage, please carefully document the extent of damage and notify the shipper of the situation. Also, please call a representative of Pacific Nanotechnology.

Chapter 2

Installation and Ethernet Configuration

This chapter describes how to install and configure the PScan2™ Controller.

2.1 Installation

Quick-check

This procedure will confirm that your PScan2™ controller has arrive safely:

1. Confirm that the voltage rating that is printed on the rear label is correct for your in-house line voltage.
2. Confirm that the enclosed power cord is correct for your in-house outlets.
3. Connect the power cord and turn-on the power switch located adjacent to the power cord inlet.
4. In addition to a low-level sound of fans operating, the controller should beep three times within 30 seconds.
5. If the beeps are not heard, call the Pacific Nanotechnology representative for further checks.

2.2 Hardware Configuration

Master Computer - Controller Ethernet Cabling

Twisted-pair connection:
The communication between the Master computer and the Pscan2™ Controller requires a 10 Mbit/sec Ethernet connection. If no hub controller is need for in-house communication between the Master Computer and other computers, then only a simple crossed-wire twisted-pair cable is required. A 6 to 15 ft. cable with male RJ-45 connectors provides the simplest and most satisfactory, bullet-proofHelpnection.

Hub-type connection:
Various low-cost 10/100 Mbit/sec hubs are available commercially from local computer stores. The user is advised to consult an expert in Ethernet communications for the particular needs at hand. For convenience, Pacific Nanotechnology offers a name-brand local hub and cabling.

Network cards and connectors:
You can use any ISA or PCI network board but we strongly recommend that you purchase a major brand board, such as 3-Com. There are so many networking products on the market, it is not easy to diagnose or anticipate all the possible problems that you may encounter. Adding to, or upgrading, your computer system requires certain knowledge and experience with computer hardware and software. If you do not have this expertise, you may want to enlist the assistance of a responsible computer professional before attempting such an upgrade.

All Pacific Nanotechnology, Inc. support notes, whether on-line or in hard copy, are designed to assist our customers in the use and maintenance of their Pacific Nanotechnology equipment. These notes are not replacements for professional technical assistance when warranted. Pacific Nanotechnology, Inc. cannot be responsible for after-sale printer or other hardware upgrades not completed by the authorized Pacific Nanotechnology, Inc. representatives.

Please send your network questions to help@pacificnanotech.com. Include a complete description of the network configuration you plan to use with your PScan2™ system.

Network software components
The Windows >95/NT network communications package offers several protocols for linking computers. A brief summary of how the communications between a Windows-based Master Computer and the DOS-based PScan2™ Controller is provided below. Please refer to the section on the DCEx™ Protocol in Chapter 5 for more detail. Understanding and implementing a communications network requires substantial expertise, and the user is advised to consult with a person with knowledge in this art.

The current configuration operates using Microsoft NetBIOS Extended User Interface (NetBEUI) protocol on both Master Workstation and Controller.

Microsoft Network Client version 3.0 for MS-DOS is installed on the Controller side and provides a file-level network access to the Master Workstation=s shared resources.

The following Network software components are required on the Master Workstation:

- Ethernet Adapter driver;
- NetBEUI protocol driver;
- Client for Microsoft Networks;
- File and Printer Sharing for Microsoft Networks service;

(**TIP: Use Settings->Control Panel->Network** to add required network components or to edit their properties. Your original Windows 95 or NT disk may be required for completing installation.)

The necessary protocol bindings are required on Master Workstation:

- NetBEUI to Ethernet Adapter;
- Client and File&Printer Sharing to NetBEUI.

The File Sharing capability on the Master Workstation must be enabled (Settings-> Control Panel-> Network-> File and Print Sharing-> “I want to be able to give others access to my files” check box checked).

Chapter 3

Description of External Signal Connections

This chapter includes specifications and signal connection instructions for PScan2™ Controller connectors located on the rear panel.

Warning: Connections that exceed any of the maximum ratings of input or output signals on the PScan2™ Controller can damage the board and the computer. In general, INPUT voltages are not to exceed +/- 20 VDC. The description of each signal includes information about maximum input ratings, if specified different than above. Pacific Nanotechnology, Inc. is NOT liable for any damages resulting from any such signal connections.

3.1 I/O Connector Pin Description

Linearizers

There are two connectors, labeled LINEARIZERS I and LINEARIZERS II, which can connect two different types of sensors, depending on the option, if any, requested by the user. For TopoMetrix scan heads that use strain-gauge sensors, the LINEARIZER II connector is active. This 9 pin sub-D connector is connected internally to a TopoMetrix strain-gauge interface board. The signal connections are as follows:

9 pin D-Sub female (Rear panel)	12 pin Molex (designators on TopoMetrix strain-gauge interface board)	
1	1	VREF -
2	2	- EDX
3	3	+ EDX
4	5	- EDY
5	6	+ EDY
6	9	+ EDZ
7	10	- EDZ
8	11	VREF +
9	4,7,8,12	NC

Scanner

The scan head, or scanner, is connected to the PSCAN2™ Controller with a 37 pin sub-D connector. The signal connections are as follows:

1 - AN. GND (for detector preamp)	**20 - DET-T/L (detector preamp, top-left)**
2 - AN. GND	**21 - DET-T/R (det. preamp, top-right)**
3 - AN. GND	**22 - DET-B/L (det. preamp, bottom-left)**
4 - GND, DCMTR	**23 - DET-B/R (det. preamp, bottom right)**
5 - EXT- (external input, common)	**24 - EXT+ (external input, +/- 10 VDC)**
6 - +15 VDC power	**25 - NC**
7 - -15 VDC power	**26 - NC**
8 - LZR-RET (laser return)	**27 - LZR-PWR (laser power, +5 VDC)**
9 - DCMTR (dc motor for probe approach)	**28 - Z-PY2 (Z piezo modulator output)**
10 - Z-RT2 (return for Z piezo modulator)	**29 - Z-PY1 (Z piezo actuator output)**
11 - Z-RT1 (return, Z piezo modulator)	**30 - Y-PIZ (Y piezo actuator output)**
12 - Y-RET (return, Y piezo)	**31 - X-PIZ (X piezo actuator output)**
13 - X-RET (return, X piezo)	**32 - N/C**
14 - N/C	**33 - N/C**
15 - N/C	**34 - N/C**
16 - N/C	**35 - Z (+) (high voltage board option)**
17 - Y (+) (high voltage board option)	**36 - Y (-)(high voltage board option)**
18 - X (+) (high voltage board option)	**37 -X (-) (high voltage board option)**
19 - GND	

Signal Access

A number of internal signals can be monitored and certain input signals can be coupled to the PScan2™ Controller via a 60 pin dual in-line connector. The signal connections are as follows:

1	**Z(SET)**	**MON1 - Set-point for Z-feedback loop**
2	**GND**	
3	**Z(POS)**	**MON2 - Error Signal, Z(ERR), with gain & filters**
4	**GND**	
5	**Z(MOD)**	**MON3 - Output signal from Frequency Synthesizer**
6	**GND**	
7	**Z(DMO)**	**MON4 - Output signal from Demodulator**
8	**GND**	
9	**Z(ERR)**	**MON5 - Comparator output signal (Z(SET)- Z(SIG))**
10	**GND**	
11	**Z(PID)**	**MON6 - Output signal from the Z-PID feedback loop**
12	**GND**	
13	**Z(SEN)**	**MON7 - Output from distance sensor along the Z-axis**
14	**GND**	
15	**Z(HGT)**	**MON8 - 1x or 3x buffered Z-PID Signal, proportional to Z height (topology)**
16	**GND**	
17	**Z(L-R)**	**MON9 - Difference signal from quadrant photodetector: Left-half minus right-half**
18	**GND**	

19	Z(T-B)	MON10 - Difference signal from quadrant photodetector: top-half minus bottom-half
20	GND	
21	X(DAC)	MON11 - Output signal for X-piezo driver
22	GND	
23	Y(DAC)	MON12 - Output signal for Y-piezo
24	GND	
25	X(SET)	MON13 - Set-point for X-linearizer feedback loop
26	GND	
27	Y(SET)	MON14 - Set-point for Y-linearizer feedback loop
28	GND	
29	X(SEN)	MON15 - Output from distance sensor along the X-axis
30	GND	
31	Y(SEN)	MON16 - Output from distance sensor along the Y-axis
32	GND	
33	X(CTL)	MON17 - Output to X-piezo from linearizer feedback loop
34	GND	
35	Y(CTL)	MON18 - Output to Y-piezo from linearizer feedback loop
36	GND	
37	Z(PIZ)	MON19 - Output signal for Z-piezo
38	GND	
39	Z(SUM)	MON20 - Sum of Photodetector quadrants
40	GND	
41	FLGSS (also pin 20, P3)	Start scan
42	DIG. OUT. (also pin 15, P1)	DIG. GND (also pin 50, P2/ 50, P3)
43	EXT MOD (was PIXCLK) (also pin 35, P1)	AN. INPUT (For Pulse-Force scanning)
44	AUX1-DAC	AN. OUTPUT
45	EXTSS (also pin 34, P1)	External start scan
46	AUX2-DAC	Auxiliary 2 Output
47	AUX1+	AN. INPUT, HI
48	AUX1-	AN. INPUT, LO
49	AUX2+	AN. INPUT, HI
50	AUX2-	AN. INPUT, LO
51	AUX1-DAC	
52	AN. GND	
53	AUX2-DAC	
54	AN. GND	
55	FLAGPT (also pin 19, P1)	Flag set/clear for each data point
56	ADC-8B	
57	+ 5 V REFB	
58	+ 5 V REFB	
59	- NC	
60	- NC	

Steppers / Digital I/O

The six stepper drivers are rated for 12 V, 0.5 A stepper motors. The pin-out description for the 50 pin dual in-line connector is presented in Appendix B.

Chapter 4

Description of Operation

This chapter contains a functional overview of the PSCAN2™ Controller and explains the operation of each functional unit comprising the PSCAN2™ Controller.

4.1 Overview

Today, Scanning Probe Microscopy encompasses a large number of techniques for positioning or scanning a small sensing element relative to the sample or specimen of interest. The sensing element may be in continually or intermittently in contact oscillating modes with the sample, or just above the sample. With most of these techniques, there are mandatory functions that the controller must perform. The principal functions are discussed below:

High speed analog signal input/output acquisition and digital control:
Most SPM techniques and scanner require repositioning the probe relative to the sample (e.g., output new X & Y analog voltages) and acquire data through one-to-several analog channels at a rate of a few times per minute to several kHz.

Feedback loops in X, Y, and Z:
Precise, reproducible and independent positioning of the tip/sensor probe is accomplished by sensing each of the relative motions and providing a feedback-loop for setting the voltage or current levels or the actuators for each direction of motion.

Signal conditioning for the probe/Z-Sensor signal:
Provision is made for Z-sensor signal that may be derived from the output of a quad-photodetector, is typically used in a light-lever AFM sensor, or a single-pole source, such as a piezo-resistive sensor probe. As required for oscillating modes, such as with vibrating cantilever or tuning fork sensing, a demodulation circuit is incorporated in order to provide a signal proportional to the amplitude of the oscillating device. The primary output signal, conditioned by gain and filtering, provides the comparison signal that is used in the Z-PID feedback loop (below). Additional output signals are conditioned for data acquisition (see detailed description below).

Z-PID feedback loop:
The conditioned Z-sensor signal is compared to a Z-SET level (positive or negative voltage) so as to generate a positive- or negative-going error signal. The error signal is further conditioned within the PID circuitry and amplified with sufficient power so as to drive a piezo actuator. And, to complete the feedback loop, the Z-piezo actuator, with sensor or sample attached at the free end, moves in the direction and to the extent so as to minimize the error signal.

Modulator and driver:
A sinusoidal frequency synthesizer with amplitude control is used to drive the oscillation sensor elements, such as cantilevers, tuning forks or optical fibers. Depending on the need, the output may be either capacitively coupled to an X, Y, or Z piezo actuator or directly driving a small bimorph piezo actuator which is mounted near the oscillating element.

Signal conditioning for Auxiliary Analog Input and Output Signals:
Often an SPM technique requires additional analog input and output signals. These signals should be buffered in order to minimize damage from excessive voltages. Also, the input signals should be operated in the differential mode so as to minimize grounding loops, and to permit the simple addition of external voltage off-set and gain circuitry.

Motor/Laser control and secondary I/O lines:
Motors are used for probe-sample approach, coarse movement of sample/stage in X, Y, and Z, and other motions particular to the SPM technique. Most SPM systems use either a small DC motor or a stepping motor for probe-sample approach and both options should be available in the controller. Although some systems use heavy-duty stepper motors for coarse movements in the three axes, others incorporate relatively small stepper motors (less than 0.5 amp per phase). The controller should have provision operating either the small steppers directly or digital lines for controller external heavy-duty stepper drivers.

For purposes of this discussion we will assume that the user is operating an AFM-type scanner using the light-lever sensing scheme (quadrant-photodetector) and low-voltage piezo actuators. Also, we will assume that the scanner has internal sensors (e.g., strain gauge or capacitance sensors) for monitoring X, Y and Z motions of the scanning tip or scanning sample. Therefore, it is possible to locate or position the tip/sensor probe absolutely in X, Y and Z by incorporating respective feedback loops to the piezo actuators.

The simplified block diagram in Appendix C-1 shows the PSCAN2™ Controller as seven main blocks or sections. The computer section (left side) generates and receives the analog and digital signals for operating the other six sections that provide the interface with the Scanner, Stage, and any Auxiliary Signal Components. The sections are presented so as to approximately match the primary functions of a SPM controller.

4.2 Detailed Block Diagram

In Appendix C-2, a more detailed block diagram is shown which present a more complete picture of how the controller functions. Please note several items:

1. Each block represents approximately one function. For brevity, only the primary functions are shown. For example, some buffering circuits are not indicated. Triangles represent one of the following amplifiers: buffering amplifier, differential amplifier, summing amplifier or power amplifier. Rectangular boxes represent a circuit function that is labeled inside the box. Where appropriate, the bit-resolutions and voltage ranges are shown.

2. The computer and analog/digital I/O support functionality is not shown. Rather, the digital output "chip select" and "switch" lines are represented as "CS-xx" and "X-xx" designators which are adjacent to the block representing the function.

3. The designators or names for the analog signal lines (also used in the schematic diagrams) are shown adjacent to the functional block. This includes the signals which can be acquired, represented by the "ADC-xx" designator which are just below the signal name.

4. Analog signal lines that can be monitored externally are represented by a triangle (buffering amplifier) and a circle enclosing a number designator.

4.3 Description

X & Y sensing and feedback loops:

a) **Non-feedback mode.** The X- & Y- DAC scan signals from the computer enter the Interface Board circuitry through their respective differential buffering amplifier, followed by an amplifier ("ZOOM") with digital gain. This signal is summed with a digitally controlled voltage source ("OFFSET") to produce a X- / Y-SET signal which can be used to control an "PI" feedback loop for each direction. If the downstream switch is set in the 1-2 position, the X-/Y-SET drives the power amplifier PA directly. The outputs X-/Y-PIZ are wired directly to their respective piezo actuators.

b) **Feedback mode,** The differential inputs from the X and Y position sensors must be preconditioned to a range of 0 - 10 V, the operating range of the summing amplifier which sums the incoming position signal with the X-/Y-SET positioning voltage to form an error signal which is conditioned by a "PI" feedback circuit. With the X-/Y-CTL switch set in the 2-3 position the resultant corrected signal drives the piezo power amplifiers.

Signal conditioning for the probe/Z-Sensor signal:

There are a number of switches and amplifiers that have the purpose of selecting the source of the incoming signal and conditioning the selected signal which will be used in the PID feedback circuitry to generate the error signal. In the case of a quadrant photodetector, there are four conditioning circuits:

1. The four signals are summed to monitor the total signal level (Z-SUM).

2. The summed right and left halves of the detector are compared (L-R) and further conditioned by gain and filtering to provide a signal (Z-LR) that represents torsion on the cantilever (as observed in frictional force microscopy).

3. The summed top and bottom halves of the detector are compared (T-B) and the resultant signal is used directly in the Z-feedback loop (SIG-IN).

4. The summed top and bottom halves of the detector are compared (T-B) and the resultant signal is used in a demodulator circuit. The demodulated output can be used in the Z-feedback loop (SIG-IN) and as an acquisition signal (Z-DEM) with some filtering.

Also, the Z-position sensor, Z-S, appropriately conditioned with gain and frequency filter selections, can be switch into Z-feedback loop input, SIG-IN for precise Z-positioning. The conditioned Z-S signal can also be acquired, say when scanning with the Z-feedback loop controlled by photodiode or external input modes.

Z-PID Feedback Loop:

In the first part of this section, the input signal, SIG-IN, is compared to a reference level, Z-SET, to generate an error signal, Z-ERR. Software-controlled switches provide the options of an inverted or non-inverted SIG-IN and a positive or negative Z-SET value. Following a gain stage, the Proportional, Integral and Derivative signals of the error signal, each with controllable gains within each section, a summed to provide a buffered signal, Z-PID. With optional gain, this signal forms the data acquisition signal, Z-HGT. The Z-PID signal is also power-amplified to form Z-PY1, the voltage for driving the Z-piezo actuator.

Modulator:

The modulator consists of a sinusoidal frequency synthesizer and a power amplifier. The synthesizer possesses a 20 MHZ clock and is capable of generating frequencies from a few kHz to several hundred MHZ, as determined by the bandwidth of the power amplifier. With 32-bit resolution, frequency increments are less than 0.005 Hz. The output amplitude ranges from 0 to +/- 10 V p-p at 9 bit resolution (about 40 mV).

Signal conditioning for Auxiliary Analog Input and Output Signals:

The auxiliary input signals, AUX1 and AUX2, are each buffered with a differential input amplifier before entering the A/D multiplex circuitry. The input voltage range is 0-10 V, and is resolved at 16-bits by the A/D converter (about 0.16 mV). The differential input allows the incoming signals to be easily inverted and offset externally.

The auxiliary output signals range from 0 - 10 V and are generated at 12-bit resolution (2.5 mV).

Motor/Laser Control and secondary I/O lines:

The motor/laser control section comprises a means for driving 6 small stepper motors, one DC motor and a switch for turning the laser on and off. The drivers for the small steppers are rated at 0.5 amp/phase and independent control lines for motor select, step size (full/half-step), direction, and current control. The output driver for the DC motor provides an output of +/- 5 VDC, at 100 - 150 mA, at 8-bit resolution (about 40 mV). The solid state switch that enables the DC motor relay also enables the laser. This prevents inadvertent turning-on of these components during computer boot-up and initialization of hardware signal states.

Schematic diagrams are provided in Appendix D, (separate volume).

Chapter 5

Data Command Exchange (DCEx™)

The purpose of the DCEx™ protocol is to provide reliable data transfer between master and the slave over the Ethernet using standard Windows drivers. This protocol is designed for both command and data exchange. The data is transferred in binary file format. The commands are issued by the presence (or absence) of specific files in a predefined "control" directory of the master computer. The slave computer checks periodically the content and status of the control directory. Whenever a new command is issued, the slave notices the presence of a "flagged" file and responds appropriately. The slave confirms its updated status by writing a status line into the .log file that may be displayed in the master program.

The network requirements, commands and command structure, and data file formats are described below and in Appendix F.

5.1 PScan2™ Scanning Probe Microscope System Configuration

System components

The PScan2™SPM System consists of:

1. Master Workstation that operates under MS Windows 95™ or MS Windows NT™ operating system and runs Application Software;
2. Controller that operates under MS-DOS 6.22 and runs Controller Software;
3. Scanner (scanhead) that is connected to Controller's Interface Board;
4. Ethernet network link between the Master Workstation and Controller that is implemented via a Twisted Pair (TP) DirectLink Ethernet cable, two regular TP cables and Ethernet TP-Hub, or a coaxial Ethernet cable and two Ethernet

network cards (10Mbps or 100Mbps) in the Master Workstation and Controller correspondingly. (See Figure 5-1 for basic system block diagram).

Network software components

The current configuration operates using Microsoft NetBIOS Extended User Interface (NetBEUI) protocol on both Master Workstation and Controller.

Microsoft Network Client version 3.0 for MS-DOS is installed on the Controller side and provides a file-level network access to the Master Workstation's shared resources.

The following Network software components are required on the Master Workstation:

- Ethernet Adapter driver (Figure 5-2)
- NetBEUI protocol driver (Figure 5-4)
- Client for Microsoft Networks (Figure 5-6)
- File and Printer Sharing for Microsoft Networks service (Figure 5-7)

(TIP: Use **Settings->Control Panel->Network** to add required network components or to edit their properties. Your original Window, '95 or NT disk may be required for competing installation.)

The necessary protocol bindings are required on Master Workstation:

- NetBEUI to Ethernet Adapter (Figure 5-3).
- Client and File&Printer Sharing to NetBEUI (Figure 5-5).

The File Sharing capability on the Master Workstation must be enabled (**Settings-> Control Panel-> Network-> File and Print Sharing->** "I want to be able to give others access to my files" check box checked - see Figure 5-8).

Data Command Exchange (DCEx™) protocol structure

The Data Command Exchange (DCEx™) protocol is an Application level protocol which can be used to send a command to the Controller, receive data from the Controller, get a message or status information from the Controller, or supply configuration parameter values to the Controller. There are four major groups of DCEx™ protocol components – Command files, Data files, Log files and one Configuration file (Appendix F). These are described below.

Commands constitute empty files (except CHANGE_FLAG) that are created by the Master Workstation and checked/deleted by the Controller. They are used to put the Controller into one of the designated functional modes, exit from a current mode (STOP_FLAG) or notify the Controller about configuration parameter value changes

(CHANGE_FLAG). The CHANGE_FLAG file contains a text string with the parameter's section name that was changed and needs to be reapplied. If this file contains more than one string, then only the Controller services the last line entry.

IMPORTANT NOTE: In order to exit from a current functional mode, the Controller needs a STOP_FLAG command. This is because mode commands cannot interrupt each other (except DCMTR_FWD and DCMTR_REV mode commands which allow for an interrupt).

Data files are created and filled by the Controller and can be read by the Application software on the Master Workstation. These Data files contain ADC measurements for oscilloscope modes, for the frequency sweep mode, for scanning mode and Red Dot alignment mode. They are stored in either binary or ASCII format, depending on volume and throughput.

The **Log** file ERROR.LOG is created and filled by the Controller and can be accessed by the Application software on the Master Workstation. It is a text file in which each line is a message string or status string from the Controller. The Controller sends an empty string when it comes to the Idle mode. The last line of ERROR.LOG represents the most recent message from the Controller.

The **Log** file LINE.LOG contains one text line with the number of scan lines for which data has already been acquired. It can be used by the Application software for scan progress monitoring and scan image data tracking.

The **Configuration** file SLAVE.INI is represented as a generic INI-file structure:

```
[SECTION1 NAME]
KEY1_NAME=KEY1_VALUE
KEY2_NAME=KEY2_VALUE
….
[SECTION2 NAME]
KEY3_NAME=KEY3_VALUE
KEY4_NAME=KEY4_VALUE
```

The section name must be in square brackets. The parameter description string starts with a key name, followed by "=" sign, then by the parameter value and ends with an "Enter". The order of keys within a section and the order of sections are not important. The detailed description of all configuration parameters, their values and related Controller hardware signals are provided in Appendix G:(Slaveini.xls and Slaveini.doc).

A typical example of a DCEx™ communication transaction

Assume that the Controller is in "Oscilloscope time"– mode and the next activity is a "Scan"-mode operation. Then the typical Application software actions would include the following:

- Issue a STOP_FLAG command to exit from the current mode (creates file "stop.flg" in Device Directory);
- Modify parameters in SLAVE_INI file, if needed
- Issue a SCAN_START command to start scan operation (creates file "scanstrt.flg" in Device Directory)
- Periodically, read text line from the LINE_LOG file and read the corresponding data set from the SCAN_DAT file, and update scan progress indicator and process/display image
- Modify parameters in the SLAVE_INI file and create a CHANGE_FLAG file in the Device Directory with a changed section name in it (one at a time), if needed, while scan operation is still in progress
- Issue a STOP_FLAG command, if needed, in order to terminate scan operation before its completion.

5.2 Commands and Controller Functional Modes

The Controller is designed to operate in one of the specific functional modes. There are currently 13 functional modes; each functional mode represents a specific task performed by the Controller. Functional mode can be either time-unlimited or time-limited. An example of a time-unlimited mode is the "Oscilloscope, time mode"; the Controller is allowed to stay in this mode as long as appropriate. An example of a time-limited mode is the "Scan Image" mode; the Controller will exit this mode as soon as the scan operation is completed.

DCEx™ commands are used to navigate the Controller through functional modes, initiate a specific operation or abort current operation (STOP_FLAG), request current status (PING_FLAG) or notify the Controller about operating parameters change (CHANGE_FILE).

In addition to 13 functional modes there are two "Standalone" modes that are designed for the Controller's network configuration and the Controller's software update. "Standalone" here means that the Controller is not connected to the Master workstation. There are two "Standalone" commands, CONFIGURE_FLAG and UPDATE_FLAG. "Standalone" commands are supplied to the Controller via floppy disk drive and are checked by the Controller only during boot up and only in standalone configuration (the Controller is not connected to the Master workstation).

5.2.1. IDLE MODE

```
Command: STOP_FLAG
Default mode for power on and reboot
Time-unlimited
Logs used: ERR_LOG, PING_LOG, PID_LOG
Data files: none
```

The Idle mode is the default mode that the Controller enters after first power up, reboot or after the STOP_FLAG command is issued. Being in this mode, the Controller polls Device Directory for the occurrence of any Command flag (command file). The following cycled order is used for commands polling:

```
Reset (RESET_FLAG);
Ping (PING_FLAG);
Change (CHANGE_FILE);
Tip retract (TIP_UP);
Change (CHANGE_FILE);
Tip approach (TIP_DOWN);
Change (CHANGE_FILE);
Red Dot alignment (REDDOT_START);
Change (CHANGE_FILE);
Scan Image (SCAN_START);
Change (CHANGE_FILE);
Oscilloscope, time mode (OSC1_START);
Change (CHANGE_FILE);
Oscilloscope, line mode (OSC2_START);
Change (CHANGE_FILE);
Frequency sweep (SWEEP_START);
Change (CHANGE_FILE);
Oscilloscope, storage mode (OSCSTO_START);
Change (CHANGE_FILE);
Stepper motor (STEPPER_START);
Change (CHANGE_FILE);
DC motor forward (DCMTR_FWD);
Change (CHANGE_FILE);
DC motor reverse (DCMTR_REV);
```

Once command flag is detected, the Controller performs an appropriate action or enters into one of the functional modes.

The CHANGE_FILE command flag is checked every time between two functional mode command flag checks. If CHANGE_FILE is detected, the parameter values from the appropriate section of the Slave.ini file are applied.

Every time the Controller completes or aborts current functional mode operation, it returns into the Idle mode and proceeds with the command polling according to the cycled order above. Let assume as an example that the Controller has just completed scan image operation, then it will enter the Idle Mode and check for the presence of the "Oscilloscope, time mode" (OSC1_START) command, then the "Oscilloscope, line mode" (OSC2_START) command and so on. Let assume further, that the Controller encounters the "Oscilloscope, line mode" (OSC2_START) command. Then the Controller would enter into the "Oscilloscope, line mode" functional mode and operate there until stop command (STOP_FLAG) is issued. When the stop command is issued, the Controller returns to the Idle mode and continues command polling with the "Frequency sweep" command checked next (see cycled polling order above).

Whenever the Controller is initialized (on power up or software reboot), it writes the Controller Software version information and the "Device Initialized" line into the log file (ERROR_LOG) and enters into the Idle mode.

5.2.2. RESET MODE

```
Command: RESET_FLAG
Time-limited
Logs used: ERR_LOG, PID_LOG
Data files: none
```

The purpose of the reset mode is to reinitialize the Controller's Interface Board and to reopen the log file (ERROR_LOG). **The Controller's computer is not reinitialized, reset, or affected by any means during this mode, nor the Controller's software is reloaded. Hardware power on reset must be used for a complete Controller system reinitialization**. This mode is designed to handle network communication failures in network link between the Controller and the Master Workstation. It is recommended that the Application Software on the Master Workstation issues "reset" command (RESET_FLAG) right after the "stop" command (STOP_FLAG) every time it is loaded.

Let's assume as an example that the Controller is in the "Image scan" mode and then suddenly the Master Workstation hangs. The Controller will then be stuck on a network I/O operation. After the Master Workstation reboots the Controller resumes a network operation and continues its functioning. All information that designated to data and log files that were open by Controller before the Master Workstation was

reboot is going nowhere and is lost. The Controller remains in the same functional mode that it was in at the moment of the Master Workstation hung up. When the Application Software issues the "stop" command, the Controller is in the Idle mode, but log messages are still going nowhere. Then the "reset" command forces the Controller to reopen the log file and the Controller is ready to proceed with operation.

Whenever the Controller services the "reset" command, it writes the Controller Software version information and the "Device Ready" line into the log file (ERROR_LOG).

5.2.3. TIP RETRACT MODE

```
Command: TIP_UP
Time-limited
Logs used: ERR_LOG, PID_LOG
Data files: none
```

This mode is used for SPM probe tip retract operation. During this operation the Controller first of all writes the "Tip Retract" line into the log file (ERROR_LOG), then performs "fast retract" by activating "fast retract" line (X5). The Controller further accesses parameter ZMTR_TIP in the Slave.ini file, section [TIP APPROACH], and uses its value for a Z-motor selection.

If Z DC motor is selected, the Controller accesses parameter DCREV_TIP and DCTIME_TIP values in the Slave.ini file, section [TIP APPROACH]. The values are used to apply a specified DC motor voltage for a specified period of time.

If one of the eight stepper motor is selected, the Controller accesses parameter DIRUP_TIP, STEPUP_TIP, PULSES_TIP and PACKET_TIP values in the Slave.ini file, section [TIP APPROACH]. The values are used to select the direction, full/half step, the number of pulses and pulse packet size for tip retraction using stepper motor. Stepper pulses are produced at 1 kHz rate; the network I/O operation (which takes additional time out of stepping) is performed only between pulse packets. Thus the PACKET_TIP value determines the actual speed of tip retraction using stepper motor.

Tip retraction can be terminated before the completion (DCTIME_TIP elapsed time or PULSES_TIP stepper pulses) by a "stop" command. The Controller checks for a STOP_FLAG at about every 100 ms time interval if a DC motor is used and after each stepper pulse packet if a stepper motor is used. When the tip retraction is completed or terminated, the Controller writes an empty line into the log file (ERROR_LOG).

5.2.4. TIP APPROACH MODE

```
Command: TIP_DOWN
Time-limited
Logs used: ERR_LOG, PID_LOG
Data files: none
```

This mode is used for SPM probe tip approach and Z-PID feedback engage operation. When the Controller enters into this mode it writes the "Tip Engage" line into the log file (ERROR_LOG). Then the Controller sets the Z PID On/Off switch (X4) into the state according to the value of the PID_ON parameter (Slave.ini file, [PID ON/OFF] section). The Controller further sets the Z-DAC output to 0Volt level that means Z piezo is fully extended. After that the Controller accesses parameters ZMTR_TIP, CH_TIP and SRF_TIP from the Slave.ini file, section [TIP APPROACH].

If the Z DC motor is selected, the Controller accesses parameter DCFWD_TIP in the Slave.ini file, section [TIP APPROACH], and uses its value for Z DC motor DAC output. Then the Controller enters into the following loop:

- Check for a STOP_FLAG; if found, then activate fast retract line (X5), set Z DC motor DAC to zero output level, set Z DAC output to +10 Volt level (Z piezo fully retracted), deactivate fast retract line (X5) and terminate current mode;
- Acquire channel set by CH_TIP parameter value and compare acquired value with the SRF_TIP value; if value is close, then set Z DC motor DAC to zero output level, activate the Z PID On/Off switch (X4 line into ON state) and complete current mode.

If one of the eight stepper motors is selected by ZMTR_TIP parameter value, the Controller accesses parameters DIRDWN_TIP, STEPDWN_TIP and CYCLES_TIP in the Slave.ini file, section [TIP APPROACH], and uses their values for stepper direction, full/half step and acquisition rate selection. Then the Controller enters into the following loop:

- Generate one pulse for the selected stepper motor
- Check for a STOP_FLAG; if found, then activate fast retract line (X5), set ZDAC output to +10 Volt level (Z piezo fully retracted), deactivate fast retract line (X5) and terminate current mode

- Acquire channel set by CH_TIP parameter value and compare acquired value with the SRF_TIP value; if value is close, then activate the Z PID On/Off switch (X4 line into ON state) and complete current mode. Repeat current step the number of CYCLES_TIP value times.

Every acquisition cycle takes approximately 15 microseconds; the CYCLES_TIP value determines the number of acquisition cycles between step pulses. Thus the CYCLES_TIP value determines the actual speed of tip approach using stepper motor.

When the "tip approach" mode is completed or terminated, the Controller writes an empty line into the log file (ERROR_LOG).

5.2.5. RED DOT ALIGNMENT MODE

```
Command: REDDOT_START
Time-unlimited
Logs used: ERR_LOG, PING_LOG, PID_LOG
Data files: REDDOT_DAT
```

The "Red Dot Alignment" mode is designated to trace the position of a reflected laser beam on a four-quadrant photo-detector (AFM application). When the Controller enters into this mode, it first writes the "Red Dot alignment" line into the log file (ERROR_LOG). Then the Controller applies parameter LR_G, LR_OFS, LR_F values from [INPUT SELECTS] section and parameter PID_POL, PID_SET, ZERR_G, Z_SET values from [Z FEEDBACK] section of the Slave.ini file. The Controller further selects T-B photo-detector signal as an input for Z feedback channel and selects to bypass the demodulator. Then the Controller enters into the following loop:

- Acquire Z_ERR, Z_LR, Z_SUM ADC input channel
- Write acquired values into the data file REDDOT_DAT starting from its zero position, data represented as an ASCII text line (comma separated)
- Check for a STOP_FLAG; if found, then output an empty line into the log file (ERROR_LOG) and terminate current mode

- Check for a CHANGE_FLAG; if found, then apply parameter values from an appropriate section of the Slave.ini file.

When the "Red Dot Alignment" mode is terminated, the Controller writes an empty line into the log file (ERROR_LOG).

5.2.6. SCAN IMAGE MODE

```
Command: SCAN_START
Time-limited
Logs used: ERR_LOG, LINE_LOG, PING_LOG, PID_LOG
Data files: SCAN_DAT (aka OSC2_DAT)
```

This mode is designed for SPM image acquisition. When the Controller enters into this mode, it first writes the "Scan Image" line into the log file (ERROR_LOG). Then the Controller opens the line log file (LINE_LOG) and outputs the "0" line into it, which means no line is scanned at that moment. The Controller further accesses parameter values in Slave.ini file, section [SCAN IMAGE], which are used for Scan Image operation. Before the actual Scan Image operation is started, the Controller applies parameter values for the following sections of the Slave.ini file: [INPUT SELECTS], [XY CONTROL], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [AUX 1&2], [LASER]. Then the Controller carries out the "slow" tip position initialization, the SPM tip is moved from its current arbitrary XY position to the scan start XY point. The tip is moved via a straight line using the number of POINTS increment with the rate of a given SCAN_RATE.

The actual Scan Image operation consists of the number of LINES alternating "forward" line scan and "reverse" line scan operations. The acquired data are transferred into the data file (SCAN_DAT) after line scan operation depending on the acquisition direction parameter DIR value. If DIR value is 0 ("forward" scan), then data are transferred only after "forward" line scan operations. If DIR value is 1 ("reverse" scan), then data are transferred only after "reverse" line scan operations. And finally, if DIR value is 2 ("forward/reverse" scan), then data are transferred after both "forward" and "reverse" line scan operations. Whenever scan line data are transferred into the data file (SCAN_DAT), the Controller increments the scan line counter and writes its value into the line log file (LINE_LOG) starting from the zero file position. Thus the ASCII text line in the line log file always represents the number of the line scan data sets in the data file (SCAN_DAT).

The Controller checks for a stop flag (STOP_FLAG) after **each** line scan operation. If the stop flag is found, the Controller writes an empty line into the log file (ERROR_LOG) and terminates Scan Image operation.

The Controller also checks for a change flag (CHANGE_FILE) indicator after each line scan. If this flag is found and indicates that [INPUT SELECTS], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [PID ON/OFF], [Z PIEZO], [AUX 1&2] or [LASER] section was modified, then the Controller applies parameter values from that section of the Slave.ini file and continues Scan Image operation. Else if the change flag indicates [SCAN IMAGE] or [XY CONTROL] modified section, the Controller writes the "Scan Image mode restarted" line into the log file (ERROR_LOG) and restarts the Scan Image operation from the very beginning.

When Scan Image operation is completed, the Controller writes an empty line into the log file (ERROR_LOG) and returns into the Idle mode.

5.2.7. OSCILLOSCOPE, TIME MODE

```
Command: OSC1_START
Time-unlimited
Logs used: ERR_LOG, PING_LOG, PID_LOG
Data files: OSC1_DAT
```

This mode is designed for 4 input channel acquisition at the real time scale. One hundred data points per channel are acquired during every TimeBase interval. Thus the time interval between two data points is equal to the TimeBase / 100. The TimeBase value can vary from 10 ms to 1000 ms. The acquired 100 point data are transferred as a whole set between every two TimeBase intervals, the time required for data transfer being lost from data acquisition.

When the Controller enters into this mode, it first writes the "Oscilloscope, time mode" line into the log file (ERROR_LOG). Before the actual data acquisition is started, the Controller applies parameter values for the following sections of the Slave.ini file: [INPUT SELECTS], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [AUX 1&2]. The Controller further accesses parameter TIME_BASE value in Slave.ini file, section [OSC TIME], and uses this value as a

TimeBase interval. After 100 data point are collected the Controller writes 100 16-bit values into the data file OSC1_DAT using binary format and starts next 100 data point acquisition. The Controller stays in the "Oscilloscope, time mode" until this mode is interrupted by a STOP_FLAG command.

It is allowed to change the TIME_BASE value during the "Oscilloscope, time mode" operation. The CHANGE_FILE command must be issued to force the Controller to apply an updated TIME_BASE value.

The Controller checks for a change flag (CHANGE_FILE) indicator after each series of data point acquisition. If this flag is found and indicates that [INPUT SELECTS], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [PID ON/OFF], [Z PIEZO], [AUX 1&2], [LASER] or [OSC TIME] section was modified, then the Controller applies parameter values from that section of the Slave.ini file and continues "Oscilloscope, time mode" operation.

5.2.8. OSCILLOSCOPE, LINE SCAN MODE

```
Command: OSC2_START
Time-unlimited
Logs used: ERR_LOG, LINE_LOG, PING_LOG, PID_LOG
Data files: OSC2_DAT (aka SCAN_DAT)
```

This mode is designed for repetitive acquisition of up to 4 selected input channels during one line of XY raster scanning. This mode is analogous to the Scan Image mode, except only one line is scanned. Data can be acquired during either forward or reverse or both directions of the line scan.

When the Controller enters into this mode, it first writes an "Oscilloscope, line mode" text string into the error log file (ERROR_LOG). Then the Controller opens the line log file (LINE_LOG) and writes a "0" into it, which means no line is scanned at that moment. The Controller then accesses parameter values in the section of the Slave.ini file called [SCAN IMAGE], which are used for line scan operation. Before the actual image scan operation is started, the Controller applies parameter values for the following sections of the Slave.ini file: [INPUT SELECTS], [XY CONTROL], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [AUX 1&2], [LASER]. Then the Controller carries out the "slow" tip position initialization, the SPM tip is moved from its current arbitrary XY position to the scan start XY point. The tip is moved via a straight line using the number of POINTS increment with the rate of a given SCAN_RATE.

The actual line scan operation consists of alternating “forward” line scan and “reverse” line scan operations. The acquired data are transferred into the data file (OSC2_DAT) after each line scan operation depending on the acquisition direction parameter DIR value. If DIR value is 0 (“forward” scan), then data are transferred only after “forward” line scan operations. If DIR value is 1 (“reverse” scan), then data are transferred only after “reverse” line scan operations. And finally, if DIR value is 2 (“forward/reverse” scan), then data are transferred after both “forward” and “reverse” line scan operations. Whenever scan line data are transferred into the data file (OSC2_DAT), the Controller increments the scan line counter and writes its value into the line log file (LINE_LOG) starting from the zero file position. Thus the ASCII text line in the line log file always represents the number of the line scan data sets in the data file (OSC2_DAT). This number can be either 0 (no data currently available), 1 (data for one line scan are collected) or 2 (data for both “forward” and “reverse” lines are collected, DIR=2). The repetitive data for each line scan operation are written to the data file (OSC2_DAT) always starting from the zero file position.

The Controller checks for a stop flag (STOP_FLAG) after **each** line scan operation. If the stop flag is found, the Controller writes an empty line into the log file (ERROR_LOG) and terminates line scan operation and returns into the Idle mode.

The Controller also checks for a change flag (CHANGE_FILE) indicator after each line scan. If this flag is found and indicates that [INPUT SELECTS], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [PID ON/OFF], [Z PIEZO], [AUX 1&2] or [LASER] section was modified, then the Controller applies parameter values from that section of the Slave.ini file and continues the “Oscilloscope, line scan” operation. Else if the change flag indicates [SCAN IMAGE] or [XY CONTROL] modified section, the Controller writes the “Line scan mode restarted” line into the log file (ERROR_LOG) and restarts the “Oscilloscope, line mode” operation from the very beginning.

5.2.9. FREQUENCY SWEEP MODE

```
Command: SWEEP_START
Time-limited
Logs used: ERR_LOG, PING_LOG, PID_LOG
Data files: SWEEP_DAT
```

This mode is designed for a 4 input channel acquisition during frequency sweep on a numerically controlled oscillator. This mode allows an acquisition of a signal frequency response in a selected frequency range.

When the Controller enters into this mode, it writes the "Oscilloscope, frequency sweep mode" line into the log file (ERROR_LOG). The Controller further accesses parameter values in the Slave.ini file, section [FREQ SWEEP], which are used for a frequency sweep operation. Before the actual frequency sweep operation is started, the Controller applies parameter values for the following sections of the Slave.ini file: [INPUT SELECTS], [XY CONTROL], [Z FEEDBACK], [DEMOD SELECTS], [AUX 1&2], [LASER]. During the actual frequency sweep operation the Controller programs the numerically controlled oscillator for 400 different frequency values evenly distributed over the frequency range determined by the FREQ_S and the FREQ_E values from the Slave.ini file, section [FREQ SWEEP]. The 400 frequency point are produced at the rate of approximately 12 ms, the overall frequency sweep duration is about 5-6 seconds. The data for 4 selected input channels are acquired for every frequency point on a "first acquire then increment" principle. Thus the settling time for every frequency point is approximately 12 ms. The values acquired for each frequency point are written by the Controller into the data file (SWEEP_DAT) using ASCII text format. Thus every line in the data file (SWEEP_DAT) contains four decimal values in ASCII text format representing ADC data for 4 input channels.

The Controller checks for a stop flag (STOP_FLAG) after each frequency point acquisition. If the stop flag is detected, the Controller writes an empty line into the log file (ERROR_LOG), terminates the "Oscilloscope, frequency sweep mode" operation and returns into the Idle mode.

The Controller also checks for a change flag (CHANGE_FILE) indicator after **each** frequency point acquisition. If this flag is detected and indicates that [PID ON/OFF], [Z PIEZO], [AUX 1&2] or [LASER] section was modified, then the Controller applies parameter values from that section of the Slave.ini file and continues the "Oscilloscope, frequency sweep mode" operation. The Controller do not take any actions if the change flag indicates [INPUT SELECTS], [FREQ SWEEP], [Z FEEDBACK], [DEMOD SELECTS] or [XY CONTROL] modified section. The "Oscilloscope, frequency sweep mode" must be restarted by the user in order for the changes in sections mentioned above to take effect.

5.2.10. OSCILLOSCOPE STORAGE MODE

```
Command: OSCSTO_START, OSCSTO_NEXT
Time-unlimited
Logs used: ERR_LOG, PING_LOG, PID_LOG
Data files: OSCSTO_DAT
```

This mode is designed for 4 input channel acquisition at the real time scale. It is analogous to the "Oscilloscope, time mode" except longer TimeBase values are used. The name "Storage" is derived from a an analogy to an electronic digital storage oscilloscope. As in the case of an electronic storage scope the "Oscilloscope storage mode" is useful for an acquisition of a "slow-changing" signal. Three hundred data point per channel is acquired during every TimeBase interval. Thus the time interval between two data points is equal to the TimeBase / 300. The TimeBase value can vary from 2000 ms to 10000 ms (2s to 10 s). The acquired data point values are transferred before the next data point is acquired. This transfer on a per-point basis allows an application on a Master Workstation to trace the data during the prolonged TimeBase interval, which may constitute from 2 to 10 second.

When the Controller enters into the given mode, it first writes the "Oscilloscope Storage mode" line into the log file (ERR_LOG). Before the actual data acquisition is started, the Controller applies parameter values for the following sections of the Slave.ini file: [INPUT SELECTS], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [AUX 1&2], [XY CONTROL]. The Controller further accesses parameter TIME_BASE and DUTY_TIME values in the Slave.ini file, section [OSC STORAGE]. The DUTY_TIME value is subtracted from the TIME_BASE value, the result is used by the Controller as a TimeBase value. The DUTY_TIME value is designated for the calibration of an "Oscilloscope storage mode". The idea is that the Controller spend some amount of time for an acquisition and data transfer and some correction of a delay between every two data point is required. The DUTY_TIME value may vary from 0 to 1900 ms.

After all 300 data point are collected the Controller waits for an OSCSTO_NEXT command before proceeding with the next 300 data point acquisition. This "hand shake" confirmation allows the synchronization of the display procedure on the Master Workstation with the data acquisition procedure on the Controller. The Controller stays in the "Oscilloscope storage mode" until this mode is interrupted by a STOP_FLAG command.

It is allowed to change the TIME_BASE value during the "Oscilloscope storage mode" operation. The CHANGE_FILE command must be issued to force the Controller to apply an updated TIME_BASE value.

The Controller checks for a change flag (CHANGE_FILE) indicator after each data point acquisition. If this flag is detected and indicates that section [INPUT SELECTS], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [PID ON/OFF], [Z PIEZO], [AUX 1&2], [LASER], [XY CONTROL] or [OSC TIME] was modified, then the Controller applies parameter values from that section of the Slave.ini file and continues "Oscilloscope storage mode" operation.

5.2.11. STEPPER MOTOR MODE

```
Command: STEPPER_START
Time-limited
Logs used: ERR_LOG
Data files: none
```

This mode is designed to operate one of eight available stepper motor. Stepper motors are driven by stepping pulses, only one stepper motor at a time can be active in current mode. Multiple stepper motors should be operated consecutively via multiple STEPPER_START commands.

When the Controller enters into the given mode, it writes the “Stepper motor” line into the log file (ERR_LOG). Then the Controller accesses the parameter MOTOR, STEP_DIR, STEP, PULSES and PACKET values from the Slave.ini file, section [STEPPERS]. The MOTOR value selects one of the eight stepper motor available, the STEP_DIR value selects either “forward” or “reverse” stepping direction, the STEP value selects either “full” or “half” step. Parameter PULSES value defines the overall number of stepping pulses to be output to the stepper motor. Stepping pulses are output by packets, the number of pulses per packet is defined by the PACKET value. The Controller performs network input/output operation only between packets, therefore the actual rotation speed of stepper motor is defined by the PACKET value. The default value of PACKET is 1.

The Controller forms 1 ms duration stepping pulses and checks for the STOP_FLAG command every time between pulse packets. If STOP_FLAG is detected, the Controller aborts current mode operation and returns to the Idle mode.

When the Controller terminates or aborts the stepper motor mode it writes an empty line into the log file (ERR_LOG).

5.2.12. DC MOTOR FORWARD MODE

```
Command: DCMTR_FWD
Time-limited
Logs used: ERR_LOG
Data files: none
```

This mode is designed for the Direct Current (DC) motor operation. The DC motor is driven by a control voltage on a Digital to Analog Converter (DAC) which may vary from –5,000 mV to +5,000 mV. Different control voltage polarity yields to different DC motor rotation direction. Thus “forward” and “reverse” DC motor direction depends on a custom hardware wiring of DC motor. The two DC motor related modes of the Controller operation allows user to define which control voltage is considered “forward” and which one is considered “reverse”.

When the Controller enters into the described mode, it writes the “DC Motor Forward” line into the log file (ERR_LOG). Then the Controller accesses the parameter DCMTR_TIME and DCMTR_FWD values from the Salve.ini file, section [DC MOTOR]. The DCMTR_TIME value specifies the duration of a DC motor action and should be a multiple of 100 ms. The DCMTR_FWD value specifies the control voltage which may vary from –5,000 mV to +5,000 mV. The polarity of the control voltage determines the direction of DC motor rotation, the amplitude determines the speed of DC motor rotation.

The Controller outputs the control voltage specified by the DCMTR_FWD value to the DC motor DAC and enters into the following cycle:

- check for a STOP_FLAG command; if detected, abort current mode operation;
- wait 100 ms and compare elapsed time with the DCMTR_TIME value; if equals, then terminate current mode operation.

When the Controller terminates or aborts the “DC Motor forward” mode operation, it sets DC motor DAC to a zero volt level and writes an empty line to the log file (ERR_LOG).

5.2.13. DC MOTOR REVERSE MODE

```
Command: DCMTR_REV
Time-limited
Logs used: ERR_LOG
Data files: none
```

This mode is designed for the Direct Current (DC) motor operation. The DC motor is driven by a control voltage on a Digital to Analog Converter (DAC) which may vary from –5,000 mV to +5,000 mV. Different control voltage polarity yields to different DC motor rotation direction. Thus "forward" and "reverse" DC motor direction depends on a custom hardware wiring of DC motor. The two DC motor related modes of the Controller operation allows user to define which control voltage is considered "forward" and which one is considered "reverse".

When the Controller enters into the described mode, it writes the "DC Motor Reverse" line into the log file (ERR_LOG). Then the Controller accesses the parameter DCMTR_TIME and DCMTR_REV values from the Slave.ini file, section [DC MOTOR]. The DCMTR_TIME value specifies the duration of a DC motor action and should be a multiple of 100 ms. The DCMTR_REV value specifies the control voltage which may vary from –5,000 mV to +5,000 mV. The polarity of the control voltage determines the direction of DC motor rotation, the amplitude determines the speed of DC motor rotation.

The Controller outputs the control voltage specified by the DCMTR_REV value to the DC motor DAC and enters into the following cycle:

- check for a STOP_FLAG command; if detected, abort current mode operation;
- wait 100 ms and compare elapsed time with the DCMTR_TIME value; if equals, then terminate current mode operation.

When the Controller terminates or aborts the "DC Motor reverse" mode operation, it sets DC motor DAC to a zero volt level and writes an empty line to the log file (ERR_LOG).

5.2.14. AUTO-CONFIGURATION STANDALONE MODE

```
Command: CONFIGURE_FLAG
Standalone Controller
Logs used: none
Data files: none
```

The auto-configuration standalone mode is designed for the Controller's network configuration. "Standalone" here means that the Controller is not connected to the Master workstation. "Standalone" commands are supplied to the Controller via floppy disk drive and are checked by the Controller only during boot up and only in standalone configuration (the Controller is not connected to the Master workstation).

The Controller attempts to connect to the Master workstation specified by the "net use" command in the "drives.bat" file and create a log file (ERR_LOG) in the Device Directory every time the Controller is reboot. If the connect fails (Ethernet cable not connected, specified Master workstation name or Device Directory name do not exist in the network or access password is invalid), the Controller checks the floppy disk drive "A:\". If the floppy disk is present in the drive, the Controller first checks for the CONFIGURE_FLAG command represented by an empty "configur.flg" file. If this command is detected, the Controller enters into the auto-configuration mode. If no CONFIGURE_FLAG command is detected, the Controller checks for the UPDATE_FLAG command represented by an empty "update.flg" file. If an UPDATE_FLAG command is detected, the Controller enters into the auto-update mode.

The Controller in the auto-configuration mode copies the "drives.bat" file from the floppy disk to the Controller's hard disk drive. If operation is completed successfully, the Controller produces the sound indication of 4 short beeps and halts the system. In case of an error the Controller produces the sound indication of 1 long beep and halts the system. The error message is output to the Controller's console.

When the Controller completes the auto-configuration mode operation, it always halts the system. The Controller must be reboot in order for the configuration changes to take effect.

5.2.15. AUTO-UPDATE STANDALONE MODE

```
Command: UPDATE_FLAG
Standalone Controller
Logs used: none
Data files: none
```

The auto-update standalone mode is designed for the Controller's software update. "Standalone" here means that the Controller is not connected to the Master workstation. "Standalone" commands are supplied to the Controller via floppy disk drive and are checked by the Controller only during boot up and only in standalone configuration (the Controller is not connected to the Master workstation).

The Controller attempts to connect to the Master workstation specified by the "net use" command in the "drives.bat" file and create a log file (ERR_LOG) in the Device Directory every time the Controller is reboot. If the connect fails (Ethernet cable not connected, specified Master workstation name or Device Directory name do not exist in the network or access password is invalid), the Controller checks the floppy disk drive "A:\". If the floppy disk is present in the drive, the Controller first checks for the CONFIGURE_FLAG command represented by an empty "configur.flg" file. If this command is detected, the Controller enters into the auto-configuration mode. If no CONFIGURE_FLAG command is detected, the Controller checks for the UPDATE_FLAG command represented by an empty "update.flg" file. If an UPDATE_FLAG command is detected, the Controller enters into the auto-update mode.

The Controller in the auto-update mode saves the current version of the Controller's software executable as a "pscan.bak" file and copies the "pscan.exe" file from the floppy disk to the Controller's hard disk drive. If operation is completed successfully, the Controller produces the sound indication of 6 short beeps and halts the system. In case of an error the Controller produces the sound indication of 1 long beep and halts the system. The error message is output to the Controller's console.

When the Controller completes the auto-update mode operation, it always halts the system. The Controller must be reboot in order for the software update to take effect.

Chapter 6

SPM-Cockpit User Interface

6.1 Introduction

As a consequence of the Master-Slave architecture, the software consists of two parts: The PScan2™ Controller software (stored and executed on the PScan2™ PC under DOS) and the Windows-based application software. Thus, we have the advantages of the Windows user interface while the performance-critical scanning tasks are operating under DOC. The Ethernet link provides an asynchronous interface which uncouples the "loose" interrupt environment of Windows from the tightly controlled timing requirements of data acquisition.

Written in industry standard Visual Basic, this package is designed to satisfy basic scanning needs as well as to provide a means for testing the controller. Source code and DLL library are provided with each unit. This gives advanced users the power to program the software for particular applications.

6.2 Description of Contents

The SPM-Cockpit User software is relatively self-explanatory. Detailed descriptions of the functions are contained in the "help" file, contained in the compiled program and outlined below.

CONTENTS OF HELP FILE:

PScan2™ SPM-Cockpit User Software

About PScan2™ SPM-CockpitAbout_PScan_SPM_Cockpit_

Tool Bar

- Open Configuration File
- Save Configuration As…
- Save Image(s) in TopoMetrix format
- Settings
- Device Directory Setup
- Ping the Controller
- Red Dot Alignment

Oscilloscope, time mode

- Oscilloscope, line mode
- Oscilloscope, frequency sweep
- Dual-trace Storage Scope
- Auto-Linearizer
- Tip Approach / Retract
- Scan Control Panel

Display Scanned Image

Menu

Menu File

- Open Configuration File
- Save Configuration As…
- Edit Configuration File
- Save Image(s) in TopoMetrix format
- Save raw scan data
- Open raw scan data
- Export Displayed Image
- Preferences
- Configuration Directory
- Image files Directory
- Settings Tabs
- Raw data Directory
- Export Image Directory
- Auto-Linearizer
- Exit

Menu Settings

Menu Device

- Directory Setup

Create Device Directory

Create Configuration Diskette

Ping

Menu Tools

Menu Display

Menu Window

Menu Help

MDI Child Windows

Red Dot Alignment

Oscilloscope, time mode

Oscilloscope, line mode

Oscilloscope, frequency sweep

Dual-trace Storage Scope

Auto-Linearizer

Tip Approach / Retract

Scan Control Panel

Display Scanned Image

Status Bar

Device Directory

Status String

Traffic Light Icon

PID state Icon

Settings Tabs

Input Selects to ADC

Z Piezo

PID On / Off

Scan Image Setup

X-Y Control

Z Feedback

Frequency Synthesizer

AUX 1&2 Outputs

Demod Selects

Laser / Motors

For the latest detailed information on these topics, please click on the “Help Topics” icon in the Help Menu of the SPM-Cockpit Program.

Appendix A: Specifications for PScan2™ Controller

This appendix lists the specifications of the PSCAN2™ Controller. These specifications are typical at 70 deg. F (20 C), unless otherwise stated. (Rev. 11/00).

Summary

General:	
Physical:	
Size	15 in. (w) x 15 in. (h) x 17 in. (d) (38.1 cm x 38.1 cm x 43.2 cm)
Weight	65 lb. (29.5 kg)
Operating:	
Voltage	115/230 VAC, 50/60 Hz
Current	0.95/0.45 Amp
Temperature	50 - 95 deg. F (10 - 35 C)
Humidity	5 - 60 % RH, non-condensing
Processing:	
	PC-based, 200 MHZ or greater
	16 Mbyte RAM
	2.1 Gbyte Hard Disk Drive
	1.44 Mbyte Floppy Drive
	SVGA video card (for diagnostics)
Connecting:	
	AC Power
	Ethernet (10/100 Mbit/sec)
	Scanner/Stage, Input/Output lines
	Linearizer Inputs (X,Y,&Z)
	Signal Access Port
	Stepper Motor Port

Primary Functions:	
Conversion range	-10 V to +10 V, or 0 to +10 V
Resolution	16 bits
Number of Input Channels	1 to 4
Sampling rate	> 20 kHz @1, 2, or 3 channel acquisition > 16 kHz @ 4 channel acquisition
Freq. Response before Demod	DC to 500 kHz, nominal
Freq. Response, Z-PID loop	DC to 20 kHz
Z-feedback Loop:	
Digitally-controlled analog	
Inputs (Internal)	4 Inputs for 4-sector Photodetector
	Tuning-fork Sensor
Inputs (External)	For other AFM sensors and STM sensing
	Range: -10 V to +10 V
	Differential, buffered input
Z-height sensor	Provision for a sensor, absolute Z-piezo motion (e.g. strain gauge) to be incorporated into a z feedback loop for absolute Z positioning Range: 0 to 10 V
	Differential, buffered input
For Oscillating Modes:	
Modulator:	
Output Waveform	Sinusoidal, digitally synthesized
Frequency Range	50 to 500 kHz
Clock frequency	20 MHZ
Frequency Resolution	32-bit (1 Hz)
Output Voltage range	0 to +/- 10 V, peak-to-peak
Output Voltage Resolution	10-bits (10 mV)
Output Voltage Resolution	10-bits (10 mV)
Output Voltage Resolution	10-bits (10 mV)
Output Voltage Resolution	10-bits (10 mV)
Output Connections	Two options, capacitive coupled to - 1) Z-piezo driver via ext. resistor 2) to an independent piezo driver
Demodulator:	
Type	Balanced Demodulator
Frequency range	50 kHz - 500 kHz
Demodulated bandwidth	DC - 20 kHz
Input gain range	1x, 2x, 3x, or 4x
Output Amplifiers:	
Z Driver (output from PID loop):	
Driver output voltage range	0 to +140 V
Frequency range	DC to 20 kHz
Noise: @ Ground	3 mV, rms, nom.

Noise: @ External +5 VDC	3 mV, rms, nom.
Instantaneous max. output current	500 mA, min.
Average continuous output current	50 mA (power supply limited)
Power Rating of output amplifier	85 watts
X & Y Scan Drivers:	
Driver output voltage range	0 V to +140 V
Frequency range	DC to 20 Hz min.
Noise: @ Ground	3 mV, rms, nom.
Noise: @ External + 5 VDC	3 mV, rms, nom.
Instantaneous max. output current	500 mA, min., per axis
Average continuous output current	50 mA per axis (power supply limited)
Power Rating of output amplifiers	85 watts

Accessory Functions:		
Analog I/O:	**Input**	**Output**
Number	2, differential, buffered	2, analog gnd
Signal level	Range: -10 V to +10 V	0 to +10 or 0 to -10 V
Resolution	1.2 mV	2.4 mV
Update rate	16 kHz min	10 kHz min.
DC motor driver:		
Use	Operate probe approach motor	
Output voltage	- 5 to + 5 VDC, 8-bit resolution	
Output current	150 mA max	
Stepping Motor Drivers:		
Use	Operate for probe approach, and/or coarse X, Y, & Z motions	
Number in controller	6 each	
Operating voltage	12 VDC, max.	
Current rating	0.50 A, max.	
Software functions	enable reduced current, set direction, step	
Options	- Chip-bypass for larger external stepper drivers	
	- 6-bit port for additional steppers or other use	

Other Features:	
Digital Flags:	
Output	FLGSS - Start Scan indicator
Output	PIXCLK - X & Y Increment indicator (optional)
Output	FLGPT - Set/clear bit to flag a data point
Input	EXTSS - External start scan
16-bit Digital I/O bus	@ 10 KHz min. update

High Voltage Option:	
Add-on board for driving tube-type piezo drivers	
Driver output voltage range	225 V to +225 V
Frequency range	DC to 20 kHz
Noise: @ Ground	10 mV, rms, nom.
@ External + 5 VDC	10 mV, rms, nom.
Instantaneous max. output current	50 mA, min.
Average continuous output current	50 mA (power supply limited)
Power Rating of output amplifier	15 watts

Signal Access Module Option:
External flat cable & BNC- type connector box to monitor more than 25 incoming, outgoing and internal signals
Connections for Digital Flags

Specifications for PScan2™ SPM Controller
Secondary Level

Primary Scanning Functions:		
Basic A/D conversion:		
Conversion range	-10 V to +10 V, or 0 to +10 V, depending on input type	
Resolution	16 bits	
Number of Input Channels	1 to 4	
Sampling rate	> 20 kHz for 1, 2, or 3 channel acquisition > 16 kHz for 4 channel acquisition	
Freq. Response before De	DC to 500 kHz, nom.	
Freq. Response, Z-PID lo	DC to 20 kHz (3 db down, double Butterworth filter)	
Input types (software selected):		
For PID loop		
Internal (designed to accommodate light-lever type sensors	- 4 Inputs for 4-sector Photodetector - For detection of vertical (topography) cantilever motions - Input: sum of top 2 quadrants minus bottom 2 quadrants - For detection of torsional (lateral friction) cantilever motions - Input: sum of right 2 quadrants minus left 2 quadrants - For alignment purposes - Input: sum of all quadrants	
External	- For other AFM sensors and STM sensing - Range: -10 V to +10 V - Differential, buffered input	
Z-sensor	- Provision for a sensor which monitors absolute Z-piezo motion (e.g. strain gauge to be incorporated into a z feedback loop for absolute Z positioning) - Differential, buffered input - Range: 0 to 10 V	
Auxiliary Inputs	- 2 each (Aux 1 and Aux 2) - Range: -10 V to +10 V - Differential, buffered inputs	
External Modulator (Pulse force)	- 0 to approx 10V; input to Z output amp; 0 - 10 kHz	

Input signals to A/D MUX	Designate Function & Signal Conditioning	Voltage Range
Z (POS)	Error Signal, Z(ERR), with gain & filters	+/- 10 V
Z (HGT)	1x or 3x buffered Z-PID Signal, proportional	+/- 10 V
Z (L-R)	Signal (left minus right) from quad photodetector	+/- 10 V
Z (SEN)	Z sensor with offset, gain & filter	+/- 10 V
AUX (IN1)	Auxiliary Input # 1	+/- 10 V
X (SEN+)	X-sensor output	0 - 10 V
Z (DEM)	Demodulated Signal, Z(DMO), with filters	0 - 10 V
Y(SEN+)	Y-sensor output	0 - 10 V
Z(ERR)	Error Signal, absolute, from cooperator	+/- 10 V
Z(SUM)	Summed Signal of quad input photodetector	0 - +10 V

AUX(IN2)	Auxiliary Input # 2	+/- 10 V
NC	Not used	+/- 10 V

*** --- Indicates a signal, suitable for acquiring images

Demodulator:	
Type	Balanced Demodulator (sometimes called synchronous demodulation, amplitude detection)
Frequency range	50 - 500 kHz
Demodulated bandwidth	DC - 20 kHz
Input sources	Demodulated oscillating probe or other external AC signal for PID feedback loop or imaging signal
Input gain range	1x, 2x, 3x, or 4x
Output signal interconnect	Z - PID feedback loop
Imaging signal	with 10 Hz, 100 Hz, 1 kHz filters, or full-bandwidth
Z - related Signal Conditioning & Control for PID feedback loop:	
Input types from	- Photodetector (top quadrants minus bottom quadrants) - Demodulated signal of Photodetector (top quadrants minus bottom quadrants) - External signal source - Demodulated signal of External signal source - Z-sensor signal - Offset - 0 to 10 VDC, 8-bit resolution - Gain: - 1 to 255, 8-bit resolution - Filter - 10 Hz, 100 Hz, 1 kHz, and full-bandwidth - Output - For Z-PID loop or Imaging signal
Signal direction	Inverted (negative-going) and non-inverted (positive-going)
Z-Set-point level	- Range: 0 to + 10 VDC or - 10 to 0 VDC, 8-bit resolution - Positive or negative level
Comparator	- 1x Summing Amplifier - Output to PID circuitry and for Imaging signal
PID Circuitry:	
Gain	1 to 255, 8-bit resolution
Proportional	1 to 255, 8-bit resolution
Integral	1 to 255, 8-bit resolution
Derivative	1 to 255, 8-bit resolution
Output	To Z-Driver Amplifier, or for Imaging signal @ 1x or 3x gain
Offset Comparator:	
Analog switch	
Offset	0 to + 10 VDC, 8-bit resolution
PID-off	Disengages PID loop but allows software-selectable Offset to set Z-output (for independent Z-piezo positioning)
Probe Retract	+ 10 VDC signal applied to Z-Driver Amplifier for rapid probe retract during initial probe approach to surface

X - Y Raster-scanning, Signal Conditioning:	
Input Types:	
a) Direct input from X & Y scan signals (generated by algorithm during acquisition or by look-u	
Signal range	0 to +10 VDC
Resolution	12-bits
Update rate	Same rate as A/D converter
b) Through "PI" feedback loop from external X&Y Sensors (Linearizer circuitry)	
Signal range	0 to +10 VDC
- Linearizer circuitry can be bypassed by on-board jumpers/switches for Direct input from X & Y DA	
Linearizer Circuitry:	
X & Y "PI" feedback loop	
Proportional	1 to 255, 8-bit resolution
Integral	1 to 255, 8-bit resolution
Offset:	0 to + 10 VDC, 8-bit resolution
Zoom:	1x to 255 x gain, 8-bit resolution
Output Amplifiers:	
Z Driver (output from PID loop):	
Driver output voltage range	-15 V to +140 V
Frequency range	DC to 20 kHz
Noise: @ Ground	3 mV, rms, nominal
@ External + 5 VD	3 mV, rms, nominal
Instantaneous max. output	500 mA, min.
Average continuous output	50 mA (power supply limited)
Power Rating of output a	85 watts
Modulator:	
Output Waveform	Sinusoidal, digitally synthesized
Frequency Range	50 to 500 kHz
Clock frequency	20 MHZ
Frequency Resolution	32-bit
Output Voltage range	0 to +/- 10 V, peak-to-peak
Output Voltage Resolution	9-bits
Output Connections	Two options, capacitive coupled to: 1) an independent piezo 2) an external resistor connected to Z-piezo driver
X & Y Drivers:	
Driver output voltage range	-15 V to +140 V
Frequency range	DC to 20 Hz min.
Noise: @ Ground	3 mV, rms, nom.
@ External + 5 VDC	3 mV, rms, nom.
Instantaneous max. output	500 mA, min., per axis
Average continuous output	50 mA per axis (power supply limited)
Power Rating (output amplifier)	85 watts

High Voltage Option:	
Add-on board	for driving tube-type piezo drivers
Driver output voltage range	-225 V to +225 V
Frequency range	DC to 20 kHz
Noise: @ Ground	10 mV, rms, nom.
@ External + 5 VDC	10 mV, rms, nom.
Instantaneous max. output	50 mA, min.
Average continuous output	50 mA (power supply limited)
Power Rating (output amplifier)	15 watts

Accessory Functions:	
Auxiliary Output signals:	
Number	2 each
Signal level	0 to +10 VDC (polarity reversed with on-board jumper)
Resolution	12-bits
Update rate	1 to 10 kHz (approximate; software dependent)
DC motor driver:	
Use	Operate probe approach motor on some SPM scanners
Output voltage range	- 5 to + 5 VDC, 8-bit resolution
Output current	150 mA max.
On/off control	software-driven relay, connected with laser on/off; set initially to "off", then "on" during initialization
Stepping Motor Drivers:	
Use	Operate miniature geared stepper motors for probe approach, and coarse X, Y, & Z motions
Number on controller	6 each
Operating voltage	12 VDC, max.
Current rating	0.50 A, max.
Software driven function	enable (reduced current with no activity), set direction, step
Options	- Driver chips may be bypassed to allow larger external stepper drivers - Additional 2 each 3-bit ports for additional steppers or other use
Input/output Flags:	
Output	FLGSS - Start Scan indicator
Output	PIXCLK - Deleted on Rev. B, NOW Ext Mod, External Modulation Input to Z output amp.
Output	FLGPT - Set/clear bit to flag a data point
Input	EXTSS - External start scan

External monitor signals (buffered):		
1	Z (SET)	Set-point for Z-feedback loop
2	Z (POS)	Error Signal, Z(ERR), with gain & filters
3	Z (MOD)	Output signal from Frequency Synthesizer
4	Z (DMO)	Output signal from Demodulator
5	Z (ERR)	Cooperator output signal (Z(SET)- Z(SIG))
6	Z (PID)	Output signal from the Z-PID feedback controller
7	Z (SEN)	Output from distance sensor along the Z-axis
8	Z (HGT)	1x or 3x buffered Z-PID Signal, proportional to Z height
9	Z (L-R)	Difference signal from quadrant photodetector: Left-half minus Right-half)

10	Z(T-B)	Difference signal from quadrant photodetector: Top-half minus bottom-half
11	X(DAC)	Output signal for X-piezo driver
12	Y(DAC)	Output signal for Y-piezo
13	X(SET)	Set-point for X-linearizer feedback loop
14	Y(SET)	Set-point for Y-linearizer feedback loop
15	X(SEN)	Output from distance sensor along the X-axis
16	Y(SEN)	Output from distance sensor along the X-axis
17	X(CTL)	Output to X-piezo from Linearizer feedback loop
18	Y(CTL)	Output to X-piezo from Linearizer feedback loop
19	Z(PIZ)	Output signal for Z-piezo
20	Z(SUM)	Sum of Photodetector quadrants
*** Digital I/O: 16-bit output bus w/ two input and 2 output control bits ***		
*** Additional 16-bit I/O bus available for external use ***		

Internal functions:	

Analog I/O board	- 8 channels differential input, 16-bit a/d , further split to 12 channels on interface board
	- 2 x 12-bit d/a, drives x&y scan piezo actuators, with 8-bit offset and 8-bit zoom on
	- Master clock using counter/timer
	  - increments memory addresses for x & y outputs
	  - latches and digitizes up to 4 channels of analog signals
	  - internal clock
	  - flags to start & stop clock
	  - keyboard (internal start scan)
	  - external start scan
	  - indicate "next pixel"
	  - Data rate (output x & y DAC, input 1 to 4 channels):
	  - 20 kHz sampling rate for 1 to 3 input channels
	  - 15 kHz sampling rate for 4 input channels

Digital I/O	- 96 bit I/O card (some lines are multiplexed on interface board) - Analog control by DAC's
Synthesizer	- 32-bit frequency generator for oscillating AFM modes - 10-bit amplitude set
Set-point	8-bit level select
PID A-loop	3 x 8-bit settings for z feedback control loop
Z signal, gain	8-bit setting to ADC for increased z detection resolution
Z signal, bandwidth	8-bit setting of z-signal to ADC for reduced bandwidth
Z-DAC	12-bit ADC for fine tip approach and indentation measurements
PI - x-loop	2 x 8-bit setting feedback parameters for x feedback loop
PI - y-loop	2 x 8-bit setting feedback parameters for y feedback loop
X DAC, dc offset	8-bit setting of scan offset in x direction
Y DAC, dc offset	8-bit setting of scan offset in y direction
X-Y zoom	8-bit setting of x-y scan range
Z sensor	2 x 8-bit setting for filter and gain of z sensor
Demod filter	3 bandwidth settings for demodulator output
Aux out 1 & 2	2 x 12 bit DAC outputs for user
DC motor	8-bit DAC for -5 to +5 v

Digital selects:	
Polarity (sensor)	selects polarity of sensor value relative to z piezo direction
Polarity (set-point)	selects polarity of set-point value relative to z piezo direction
Laser on/off	switch for laser diode
Tip retract (1-12 gain)	enables rapid tip retraction during engagement of feedback
Z-offset	enables z-offset dac
3x - Z gain reduction	gain switch for decreasing z range
Detection mode	selects dc or oscillating scanning modes
Ext/t-b switch	selects photodetector sensing or external sensor for feedback
Gain for Demod: 2x	demodulator gain setting
Gain for Demod: 3x	demodulator gain setting
Gain for Demod: 4x	demodulator gain setting
Bandwidth (10 Hz)	bandwidth reduction from z sensor
Bandwidth (100 Hz)	bandwidth reduction from z sensor
Bandwidth (1000 Hz)	bandwidth reduction from z sensor
Bandwidth (10 Hz)	bandwidth reduction for lateral force measurements
Bandwidth (100 Hz)	bandwidth reduction for lateral force measurements
Bandwidth (1000 Hz)	bandwidth reduction for lateral force measurements
Bandwidth (10 Hz)	bandwidth reduction from demodulator
Bandwidth (100 Hz)	bandwidth reduction from demodulator
Bandwidth (1000 Hz)	bandwidth reduction from demodulator
ADC mode select	selects 4 of 8 possible input sources
off (PIDOUT)	disconnect PID output from z feedback for indentation measurements
Stepper selects	enable, and set direction & increment for 6 low current stepper motor
Digital I/O	16-bit bus w/ two input and 2 output control bits
Additional 16 bits	available for external use (on analog I/O board)
Switches:	
External	main power on/off
Jumper selects	linearizer on/off select; normally software selectable
Indicators:	
	- for +5 VDC, +12 VDC, +/- 15 VDC, +140 VDC power supplies, on interface board
Power supplies:	
Internal (from slave computer)	+12 VDC filtered for steppers + 5 VDC filtered for digital circuits
Internal (add-on linear power supplies. In Controller CPU box)	- low voltage: +/- 15 VDC - high voltage: + 140 VDC (adjustable, 125 - 140 VDC)

Appendix B: Connectors and Pin Assignments (Internal and External)

CONNECTORS FOR INTERFACE BOARD			
NUMBER	**TYPE**	**USE**	**LOCATION**
1	50 PIN MALE HEADER	DIGITAL SIGNAL TRANSFER BETWEEN SLAVE CPU & INTERFACE	ON TOP
2	10 PIN .156 WALDOM, MALE	LO VOLTAGE POWER BETWEEN SLAVE CPU & INTERFACE BOARD	ON TOP
3	50 PIN MALE HEADER	STEPPER MOTOR & DC MOTOR OUTPUTS	AT REAR
4	60 PIN MALE HEADER	SIGNAL MONITORS AND EXT START (SIGNAL ACCESS MODULE)	AT REAR
5A	15 PIN SUB-D, FEMALE	CONNECTOR AT SCANNER HEAD	AT SCANNER
5B	25 PIN SUB-D, FEMALE	PARTIAL CONNECTION TO SCANNER HEAD (LOW VOLTAGE)	INTERNAL
5C	37 PIN SUB-D, FEMALE	MAIN CONNECTION TO SCANNER HEAD (LOW & HIGH VOLTAGE)	AT REAR
6	37 PIN SUB-D, FEMALE	ANALOG SIGNAL TRANSFER BETWEEN CPU & INTERFACE BOARD	ON TOP
7	50 PIN MALE HEADER	DIGITAL SIGNAL TRANSFER BETWEEN CPU & INTERFACE BOARD	ON TOP
8	6 PIN 0.156 WALD, MALE	HV POWER TRANSFER BETWEEN CPU & INTERFACE BOARD	ON TOP
9	10 PIN 0.120 MOL PKT HDR	X, Y, & Z SENSOR INTERFACE BOARD	ON TOP

60-PIN CON FOR ANALOG SIGNALS, INPUTS AND/OR MONITOR		
Input signals to A/D MUX	**Function & Signal Conditioning**	**Range**
Z (POS)	ERROR SIGNAL, Z(ERR), WITH GAIN & FILTERS	+/- 10 V
Z (HGT)	1X OR 3X BUFFERED Z-PID SIGNAL, PROPORTIONAL	+/- 10 V
Z (L-R)	SIGNAL (LEFT MINUS RIGHT) FROM QUAD INPUT PHOTODETECTOR	+/- 10 V
Z (SEN)	Z SENSOR WITH OFFSET, GAIN & FILTER	+/- 10 V
AUX (IN1)	AUXILIARY INPUT # 1	+/- 10 V
X (SEN+)	X-SENSOR OUTPUT	0 - 10 V
Z (DEM)	DEMODULATED SIGNAL, Z(DMO), WITH FILTERS	0 - 10 V
Y (SEN+)	Y-SENSOR OUTPUT	0 - 10 V

Z (ERR)	ERROR SIGNAL, ABSOLUTE, FROM COOPERATOR	+/- 10 V
Z (SUM)	SUMMED SIGNAL OF QUAD INPUT PHOTODETECTOR	0 - 10 V
AUX (IN2)	AUXILIARY INPUT # 2	+/- 10 V
NC	NOT USED	+/- 10 V

Number	**External monitor signals (buffered):**	**Function & Signal Conditioning**
1	**Z (SET)**	SET-POINT FOR Z-FEEDBACK LOOP
2	**Z (POS)**	ERROR SIGNAL, Z(ERR), WITH GAIN & FILTERS
3	**Z (MOD)**	OUTPUT SIGNAL FROM FREQUENCY SYNTHESIZER
4	**Z (DMO)**	OUTPUT SIGNAL FROM DEMODULATOR
5	**Z (ERR)**	COOPERATOR OUTPUT SIGNAL (Z(SET)- Z(SIG))
6	**Z (PID)**	OUTPUT SIGNAL FROM THE Z-PID FEEDBACK CONTROLLER
7	**Z (SEN)**	OUTPUT FROM DISTANCE SENSOR ALONG THE Z-AXIS
8	**Z (HGT)**	1X OR 3X BUFFERED Z-PID SIGNAL, PROPORTIONAL TO Z HEIGHT
9	**Z (L-R)**	DIFFERENCE SIGNAL FROM QUADRANT PHOTODETECTOR: LEFT-HALF MINUS RIGHT
10	**Z (T-B)**	DIFFERENCE SIGNAL FROM QUADRANT PHOTODETECTOR: TOP-HALF MINUS BOTTOM
11	**X(DAC)**	OUTPUT SIGNAL FOR X-PIEZO DRIVER
12	**Y(DAC)**	OUTPUT SIGNAL FOR Y-PIEZO
13	**X(SET)**	SET-POINT FOR X-LINEARIZER FEEDBACK LOOP
14	**Y(SET)**	SET-POINT FOR Y-LINEARIZER FEEDBACK LOOP
15	**X(SEN)**	OUTPUT FROM DISTANCE SENSOR ALONG THE X-AXIS
16	**Y(SEN)**	OUTPUT FROM DISTANCE SENSOR ALONG THE Y-AXIS
17	**X(CTL)**	OUTPUT TO X-PIEZO FROM LINEARIZER FEEDBACK LOOP
18	**Y(CTL)**	OUTPUT TO Y-PIEZO FROM LINEARIZER FEEDBACK LOOP
19	**Z(PIZ)**	OUTPUT SIGNAL FOR Z-PIEZO
20	**Z(SUM)**	SUM OF PHOTODETECTOR QUADRANTS
	OUTPUT - FLGSS	START SCAN
	OUTPUT - FLGPT	SET/CLEAR BIT TO FLAG DATA POINT
	INPUT - EXTSS	EXTERNAL START SCAN
	INPUT - EXT-MOD	EXTERNAL MODULATOR INPUT (E.G., FOR PULSE FORCE MODE), REV. B (WAS PIXCLK, REV. A, SPECIFICATIONS - PSCAN2™ CONTROLLER INTERFACE BOARD)

INTERNAL FUNCTIONS WITH CONNECTOR AND PIN ASSIGNMENTS:

ANALOG I/O BOARD
- CONNECTOR #6: 37-PIN SUB-D, FEMALE
  - PINOUTS:

1 ALH0	ADC1	
2 ALH1	ADC2	
3 ALH2	ADC3	
4 ALH3	ADC4	
5 ALH4	ADC5	
6 ALH5	ADC6	
7 ALH6	ADC7	
8 ALH7	ADC8	
9 A.GND	A.GND	
10 A.GND	A.GND	
11 V.REF	N/C	
12 EXT.REF 1	N/C	
13 +12 V	N/C	
14 A. GND	A.GND	
15 D.GND	D.GND	PIN 42, P5
16 COUT 0	N/C	
17 EXTTRG	N/C	
18 N/C	N/C	
19 +5 V	N/C	
20 ALL0	LOCAL GND	
21 ALL1	LOCAL GND	
22 ALL2	LOCAL GND	
23 ALL3	LOCAL GND	
24 ALL4	LOCAL GND	
25 ALL6	LOCAL GND	
26 ALL7	LOCAL GND	
27 ALL8	LOCAL GND	
28 A.GND	X(DAC-)	
29 A.GND	Y(DAC-)	
30 AO1	X(DAC+)	
31 EXT.REF 2	N/C	
32 AO2	Y(DAC+)	
33 GATE 0	N/C	
34 GATE 1	EXTSS	PIN 45, P5
35 COUT 1	EXT MOD	PIN 43, P5
36 N/C	N/C	
37 EXTCLK	N/C	

96-BIT DIGITAL I/O BOARD:

CONNECTORS # 1 & # 7: 50-PIN HEADERS, MALE

CONTROL LINES:

GROUND FROM I/O BOARD	LOCAL GND	PIN # 50			
+ 5 VDC FROM I/O/ BOARD	N/C	PIN # 49			
DATA BUS, LSB	D00 A00	1.00	A#1	# 1	A0/32
DATA BUS	D01 A01	1.00	A#1	# 1	A1/31
DATA BUS	D02 A02	1.00	A#1	# 1	A2/30
DATA BUS	D03 A03	1.00	A#1	# 1	A3/29
DATA BUS	D04 A04	1.00	A#1	# 1	A4/28
DATA BUS	D05 A05	1.00	A#1	# 1	A5/27
DATA BUS	D06 A06	1.00	A#1	# 1	A6/26
DATA BUS	D07 A07	1.00	A#1	# 1	A7/25

		8.00			
DATA BUS	D08 A08	1.00	B#1	# 1	B0/40
DATA BUS	D09 A09	1.00	B#1	# 1	B1/39
DATA BUS	D10 A10	1.00	B#1	# 1	B2/38
DATA BUS	D11 A11	1.00	B#1	# 1	B3/37
DATA BUS	D13 A12	1.00	B#1	# 1	B4/36
DATA BUS	D14 A13	1.00	B#1	# 1	B5/35
DATA BUS	D15 A14	1.00	B#1	# 1	B6/34
DATA BUS, MSB	D16 A15	1.00	B#1	# 1	B7/33

		8.00			
WRITE TO CHIP	WR A16	1.00	C#1	# 1	C0/48
A/B CHIP SELECT	A/B A17	1.00	C#1	# 1	C1/47
AD7008, SYNTHESIZER:	TC0 A18	1.00	C#1	# 1	C2/46
	TC1 A19	1.00	C#1	# 1	C3/45
	TC2 A20	1.00	C#1	# 1	C4/44
	TC3 A21	1.00	C#1	# 1	C5/43
	LOAD A22	1.00	C#1	# 1	C6/42
	RESET A23	1.00	C#1	# 1	C7/41

		8.00			

CHIP SELECT LINES:

SYNTHESIZER (AD7008)	CS0	1.00	A#2	# 1	A0/08
Z-SET, PID COMPARATOR/ Z(MTR)	CS1	1.00	A#2	# 1	A1/07
Z(I,G), PID LOOP	CS2	1.00	A#2	# 1	A2/06
Z(P,D), PID LOOP	CS3	1.00	A#2	# 1	A3/05
STP (CLK,SET), STEPPER SELECTS	CS4	1.00	A#2	# 1	A4/04
STP (SS, DIR), STEPPER SELECTS	CS5	1.00	A#2	# 1	A5/03
X(P,I), X FEEDBACK LOOP	CS6	1.00	A#2	# 1	A6/02
Y(P,I) Y FEEDBACK LOOP	CS7	1.00	A#2	# 1	A7/01

		8.00			
ZSEN (O,G)	CS8	1.00	B#2	# 1	B0/16
UNASSIGNED	CS9	1.00	B#2	# 1	B1/15
ZDAC	CS10	1.00	B#2	# 1	B2/14

AUX1	CS11	1.00	B#2	#1	B3/13
AUX2	CS12	1.00	B#2	#1	B4/12
XZM (O (CS13 NOT DESIGNATED)	CS14	1.00	B#2	#1	B5/11
YZM(O,G)	CS15	1.00	B#2	#1	B6/10
L-R (O,G)	CS16	1.00	B#2	#1	B7/09

		8.00			
"X" LINES					
POLARITY (ZSIG NONINVERTPOL-SEN)	X0	1.00	C#2	#1	C0/24
POLARITY (ZSET NONINVERTPOL-SET)	X1	1.00	C#2	#1	C1/23
UNASSIGNED	X2	1.00	C#2	#1	C2/22
HI FOR LASER ON (LASER)	X3	1.00	C#2	#1	C3/21
HI FOR OPEN LOOP (OPEN LOOP)	X4	1.00	C#2	#1	C4/20
HI TO RETRACT (RETRACT)	X5	1.00	C#2	#1	C5/19
LO FOR 3X - Z GAIN (Z-ADC 3X)	X6	1.00	C#2	#1	C6/18
LO FOR DEMOD BYPASS (DEMOD-BP)	X7	1.00	C#2	#1	C7/17

		8.00			
GROUND FROM I/O BOARD	LOCAL GND		PIN # 50		
+ 5 VDC FROM I/O/ BOARD	N/C		PIN # 49		
EXT/T-B SWITCH, LO FOR EXT	X8	1.00	A#3	#7	A0/32
LO FOR DEMOD X2, GAIN DEMOD 2X	X9	1.00	A#3	#7	A1/31
LO FOR DEMOD X3, GAIN DEMOD 3X	X10	1.00	A#3	#7	A2/30
LO FOR DEMOD X4, GAIN DEMOD 4X	X11	1.00	A#3	#7	A3/29
LO TO SELECT Z-SEN	X12	1.00	A#3	#7	A4/28
SPARE	X13	1.00	A#3	#7	A5/27
LO FOR BWIDTH, Z-SENSOR ZSBW=10	X14	1.00	A#3	#7	A6/26
LO FOR BWIDTH, Z-SENSOR ZSBW=100	X15	1.00	A#3	#7	A7/25

		8.00			
LO FOR BWIDTH, Z-SENSOR 1, ZSBW=1000	X16	1.00	B#3	#7	B0/40
UNASSIGNED	X17	1.00	B#3	#7	B1/39
LO FOR BDWIDTH , Z-POS 1X PBW=10	X18	1.00	B#3	#7	B2/38
LO FOR BDWIDTH , Z-POS 1X PBW=100	X19	1.00	B#3	#7	B3/37
LO FOR BDWIDTH , Z-POS 1X PBW=1000	X20	1.00	B#3	#7	B4/36
UNASSIGNED	X21	1.00	B#3	#7	B5/35
LO FOR BWIDTH, L-R ADC LRBW=10	X22	1.00	B#3	#7	B6/34
LO FOR BWIDTH, L-R ADC LRBW=100	X23	1.00	B#3	#7	B7/33

		8.00			
LO FOR BWIDTH, L-R ADC LRBW=1000	X24	1.00	C#3	#7	C0/48
UNASSIGNED	X25	1.00	C#3	#7	C1/47
LO FOR BWIDTH, DEMOD 10, ZDBW=10	X26	1.00	C#3	#7	C2/46
LO FOR BWIDTH, DEMOD 10, ZDBW=100	X27	1.00	C#3	#7	C3/45
LO FOR BWIDTH, DEMOD 10, ZDBW=1000	X28	1.00	C#3	#7	C4/44

LO FOR A CHAN OF ADC#5	X29	1.00	C#3	#7	C5/43
LO FOR A CHAN OF ADC#6	X30	1.00	C#3	#7	C6/42
LO FOR A CHAN OF ADC#7	X31	1.00	C#3	#7	C7/41

		8.00			
RESERVED, 16 BIT I/0 BUS	N/C	1.00	A#4	#7	A0/08
RESERVED, 16 BIT I/0 BUS	N/C	1.00	A#4	#7	A1/07
RESERVED, 16 BIT I/0 BUS	N/C	1.00	A#4	#7	A2/06
RESERVED, 16 BIT I/0 BUS	N/C	1.00	A#4	#7	A3/05
RESERVED, 16 BIT I/0 BUS	N/C	1.00	A#4	#7	A4/04
RESERVED, 16 BIT I/0 BUS	N/C	1.00	A#4	#7	A5/03
RESERVED, 16 BIT I/0 BUS	N/C	1.00	A#4	#7	A6/02
RESERVED, 16 BIT I/0 BUS	N/C	1.00	A#4	#7	A7/01

		8.00			
RESERVED, 16 BIT I/0 BUS	N/C	1.00	B#4	#7	B0/16
RESERVED, 16 BIT I/0 BUS	N/C	1.00	B#4	#7	B1/15
RESERVED, 16 BIT I/0 BUS	N/C	1.00	B#4	#7	B2/14
RESERVED, 16 BIT I/0 BUS	N/C	1.00	B#4	#7	B3/13
RESERVED, 16 BIT I/0 BUS	N/C	1.00	B#4	#7	B4/12
RESERVED, 16 BIT I/0 BUS	N/C	1.00	B#4	#7	B5/11
RESERVED, 16 BIT I/0 BUS	N/C	1.00	B#4	#7	B6/10
RESERVED, 16 BIT I/0 BUS	N/C	1.00	B#4	#7	B7/09

		8.00			
SPARE BIT, INPUT		1.00	C#4	#7	C0/24
SPARE BIT, INPUT		1.00	C#4	#7	C1/23
SPARE BIT, INPUT		1.00	C#4	#7	C3/22
SPARE BIT, INPUT		1.00	C#4	#7	C4/21
FLAGSS, FLAG FOR START (LAST NIBBLE –OUT)		1.00	C#4	#7	C5/20
FLAGPT, SET/CLEAR PER DATA POINT		1.00	C#4	#7	C6/19
LO FOR A CHAN OF ADC#8	X32	1.00	C#4	#7	C7/18
LO FOR DEMOD ENABLE	X33	1.00	C#4	#7	C7/17

		4.00			
		==========			
	ASSIGNED:	76.00 OF 96 BITS			

STEPPER MOTORS

CONNECTOR # 3: 50 PIN HEADER, MALE

STEPPER LETTER	SIGNAL	USE	SIGNAL	USE
"A"	1	PHASE C1	2	PHASE C2
	3	PHASE C3	4	PHASE C4
"B"	5	PHASE C1	6	PHASE C2
	7	PHASE C3	8	PHASE C4
"C"	9	PHASE C1	10	PHASE C2
	11	PHASE C3	12	PHASE C4
"D"	13	PHASE C1	14	PHASE C2
	15	PHASE C3	16	PHASE C4
"E"	17	PHASE C1	18	PHASE C2
	19	PHASE C3	20	PHASE C4
"F"	21	PHASE C1	22	PHASE L2
	23	PHASE L3	24	PHASE L4
"G"	25	CLK"X"CLOCK STE	26	HS"X" HALF-STEP
	27	ST"X" SET BIAS	28	DIR"X"DIRECTION
"H"	29	CLK"X"CLOCK STE	30	HS"X" HALF-STEP
	31	ST"X" SET BIAS	32	DIR"X"DIRECTION
	33	+5 VDC	34	GND
	35	N/C	36	GND
	37	N/C	38	N/C
	39	N/C	40	N/C
	41	N/C	42	N/C
	43	N/C	44	N/C
	45	N/C	46	N/C
	47	N/C	48	N/C
	49	N/C	50	N/C

ANALOG SIGNALS, INPUTS AND/OR MONITOR

CONNECTOR # 4: 60-PIN MALE HEADER

INPUT SIGNALS TO ANALOG FUNCTION & SIGNAL CONDITIONING	MUX. CHAN	RANGE
Z (POS)	ERROR SIGNAL, Z(ERR), WITH GAIN & FILTERS	+/-10 V
Z (HGT)	1X OR 3X BUFFERED Z-PID SIGNAL, PROPORTIONAL	+/-10 V
Z (L-R)	SIGNAL (LEFT MINUS RIGHT) FROM QUAD INPUT PHOTODETECTOR	+/-10 V
Z (SEN)	Z SENSOR WITH OFFSET, GAIN & FILTER	+/-10 V
AUX(IN1	AUXILIARY INPUT # 1	+/-10 V
X(SEN+)	X-SENSOR OUTPUT	0 -+10 V
Z(DEM)	DEMODULATED SIGNAL, Z(DMO), WITH FILTERS	0 -+10 V
Y(SEN+)	Y-SENSOR OUTPUT	0 -+10 V
Z(ERR)	ERROR SIGNAL, ABSOLUTE, FROM COMPARATOR	+/-10 V
Z(SUM)	SUMMED SIGNAL OF QUAD INPUT PHOTODETECTOR	0 -+10 V
AUX(IN2)	AUXILIARY INPUT # 2	+/-10 V
NC	NOT USED	+/-10 V

OUTPUT MONITOR POINTS

1	Z(SET)	SET-POINT FOR Z-FEEDBACK LOOP
2	Z(POS)	ERROR SIGNAL, Z(ERR), WITH GAIN & FILTERS
3	Z(MOD)	OUTPUT SIGNAL FROM FREQUENCY SYNTHESIZER
4	Z(DMO)	OUTPUT SIGNAL FROM DEMODULATOR
5	Z(ERR)	COMPARATOR OUTPUT SIGNAL (Z(SET)- Z(SIG))
6	Z(PID)	OUTPUT SIGNAL FROM THE Z-PID FEEDBACK CONTROLLER
7	Z(SEN)	OUTPUT FROM DISTANCE SENSOR ALONG THE Z-AXIS
8	Z(HGT)	1X OR 3X BUFFERED Z-PID SIGNAL, PROPORTIONAL TO Z HEIGHT
9	Z(L-R)	DIFFERENCE SIGNAL FROM QUADRANT PHOTODETECTOR: LEFT-HALF MINUS RIGHT-HALF)
10	Z(T-B)	DIFFERENCE SIGNAL FROM QUADRANT PHOTODETECTOR: TOP-HALF MINUS BOTTOM-HALF
11	X(DAC)	OUTPUT SIGNAL FOR X-PIEZO DRIVER
12	Y(DAC)	OUTPUT SIGNAL FOR Y-PIEZO
13	X(SET)	SET-POINT FOR X-LINEARIZER FEEDBACK LOOP
14	Y(SET)	SET-POINT FOR Y-LINEARIZER FEEDBACK LOOP
15	X(SEN)	OUTPUT FROM DISTANCE SENSOR ALONG THE X-AXIS
16	Y(SEN)	OUTPUT FROM DISTANCE SENSOR ALONG THE Y-AXIS
17	X(CTL)	OUTPUT TO X-PIEZO FROM LINEARIZER FEEDBACK LOOP
18	Y(CTL)	OUTPUT TO Y-PIEZO FROM LINEARIZER FEEDBACK LOOP
19	Z(PIZ)	OUTPUT SIGNAL FOR Z-PIEZO
20	Z(SUM)	SUM OF PHOTODETECTOR QUADRANTS

OTHER DIGITAL SIGNALS

INPUT		
	EXTSS	EXTERNAL START SCAN
OUTPUT		
	FLGSS	START SCAN
	PIXCLK	CLOCK (AVAILABLE AS OPTION)
	FLAGPT	FLAG SET/CLEAR FOR EACH DATA POINT

CONNECTOR # 4: 60 PIN HEADER, MALE

SIGNAL DESIGNATOR	SIGNAL USE/SOURCE	TYPE	SIGNAL DESIGNATOR	SIGNAL USE/SOURCE	TYPE
1	Z(SET)MON1	MONITOR	2	GND	MONITOR
3	Z(POS)MON2	MONITOR	4	GND	MONITOR
5	Z(MOD)MON3	MONITOR	6	GND	MONITOR
7	Z(DMO)MON4	MONITOR	8	GND	MONITOR
9	Z(ERR)MON5	MONITOR	10	GND	MONITOR
11	Z(PID)MON6	MONITOR	12	GND	MONITOR
13	Z(SEN)MON7	MONITOR	14	GND	MONITOR
15	Z(HGT)MON8	MONITOR	16	GND	MONITOR
17	Z(L-R)MON9	MONITOR	18	GND	MONITOR
19	Z(T-B)MON10	MONITOR	20	GND	MONITOR
21	X(DAC)MON11	MONITOR	22	GND	MONITOR
23	Y(DAC)MON12	MONITOR	24	GND	MONITOR
25	X(SET)MON13	MONITOR	26	GND	MONITOR
27	Y(SET)MON14	MONITOR	28	GND	MONITOR
29	X(SEN)MON15	MONITOR	30	GND	MONITOR
31	Y(SEN)MON16	MONITOR	32	GND	MONITOR
33	X(CTL)MON17	MONITOR	34	GND	MONITOR
35	Y(CTL)MON18	MONITOR	36	GND	MONITOR
37	Z(PIZ)MON19	MONITOR	38	GND	MONITOR
39	Z(SUM)MON20	MONITOR	40	GND	MONITOR
41	FLGSS PIN 20 - PIN 50, P2 DIG. GND - PIN 50, P3 DIG. GND	- DIG. OUTPUT	42	DIG. O - PIN 15, PDIG. GND	
43	EXT MOD - PIN 35, P6	- AN. INPUT	44	AUX1-DAC	AN.OUTPUT
45	EXTSS - PIN 34, P6 DIG. INPUT		46	AUX2-DAC	AN.OUTPUT
47	AUX1+	AN.INPUT, HI	48	AUX1-	AN.INPUT, LO
49	AUX2+	AN.INPUT, HI	50	AUX2-	AN.INPUT, LO
51	AUX1-DAC		52	AN. GND	
53	AUX2-DAC		54	AN. GND	
55	FLAGPT PIN 19, P1 FLAG DATA POINT		56	ADC-8B	
57	+ 5 V REFB		58	+ 5 V REFB	
59	NC		60	NC	

POWER IN, LOW VOLTAGE					
CONNECTOR # 2: 10 PIN 0.156 WALDOM HEADER MALE					
1	NC	***	6	NC	***
2	+ 12 VDC FROM COMPUTER	YEL	7	GROUND FROM COMPUTER PS	BLK
3	+ 5 VDC FROM COMPUTER POWER SUPPLY	RED	8	GROUND FROM COMPUTER PS	BLK
4	+ 15 VDC	ORG	9	ANALOG GND	BLK
5	- 15 VDC	GRN	10	CHASSIS GROUND	

SPM SCANNING HEAD			
CONNECTOR # 5B: 25 PIN SUB-D, FEMALE			
1	AN. GND	14	DET-T/
2	AN. GND	15	DET-T/
3	AN. GND	16	DET-B/
4	GND, DCMTR	17	DET-B/
5	EXT-	18	EXT+
6	+15 VDC	19	NC
7	-15 VDC	20	NC
8	LZR-RET	21	LZR-PWR
9	DCMTR	22	Z-PY2
10	Z-RT2	23	Z-PY1
11	Z-RT1	24	Y-PIZ
12	Y-RET	25	X-PIZ
13	X-RET		
CONNECTOR 5C @ REAR PANEL: 37 PIN SUB-D, FEMALE			
1	AN. GND (FOR DETECTOR PREAMP)	20	DET-T/L (DETECTOR PREAMP, TOP-LEFT)
2	AN. GND	21	DET-T/R (DET. PREAMP, TOP-RIGHT)
3	AN. GND	22	DET-B/L (DET. PREAMP, BOTTOM-LEFT)
4	GND, DCMTR	23	DET-B/R (DET. PREAMP, BOTTOM RIGHT)
5	EXT- (EXTERNAL INPUT, COMMON)	24	EXT+ (EXTERNAL INPUT, +/- 10 VDC)
6	+15 VDC POWER	25	NC
7	-15 VDC POWER	26	NC
8	LZR-RET (LASER RETURN)	27	LZR-PWR (LASER POWER, +5 VDC)
9	DCMTR (DC MOTOR FOR PROBE APPROACH)	28	Z-PY2 (Z PIEZO MODULATOR OUTPUT)
10	Z-RT2 (RETURN FOR Z PIEZO MODULATOR)	29	Z-PY1 (Z PIEZO ACTUATOR OUTPUT)
11	Z-RT1 (RETURN, Z PIEZO MODULATOR)	30	Y-PIZ (Y PIEZO ACTUATOR OUTPUT)
12	Y-RET (RETURN, Y PIEZO)	31	X-PIZ (X PIEZO ACTUATOR OUTPUT)
13	X-RET (RETURN, X PIEZO)	32	N/C
14	N/C	33	N/C
15	N/C	34	N/C
16	N/C	35	Z(+) (HIGH VOLTAGE BOARD OPTION)
17	Y(+) (HIGH VOLTAGE BOARD OPTION)	36	Y(-) (HIGH VOLTAGE BOARD OPTION)
18	X(+) (HIGH VOLTAGE BOARD OPTION)	37	X(-) (HIGH VOLTAGE BOARD OPTION)
19	GND		

CONNECTOR 5A FOR PACIFIC NANOTECHNOLOGY SCAN HEAD: 15 PIN COMPACT "D", FEMALE:					
1	Y-PIZ	**6**	Y-RET	**11**	AN. GND
2	X-PIZ	**7**	X-RET	**12**	DET-T/L
3	Z-PY1	**8**	Z-RT1	**13**	DET-T/R
4	+ 15 VDC	**9**	Z-PY2	**14**	DET-B/L
5	- 15 VDC	**10**	LZR-PWR	**15**	DET-B/R

X-Y-Z SENSOR BOARD

CONNECTOR # 9: 10 PIN SINGLE-ROW MOLEX 0.120 POCKET HEADER, TYPE G, MALE

1	X(SEN+)		**6**	ZS	GND
2	Y(SEN-)	GND	**7**	+ 15 VDC	
3	Y(SEN+)		**8**	POWER	GND
4	Y(SEN-)	GND	**9**	- 15 VDC	
5	ZS+		**10**	NC	

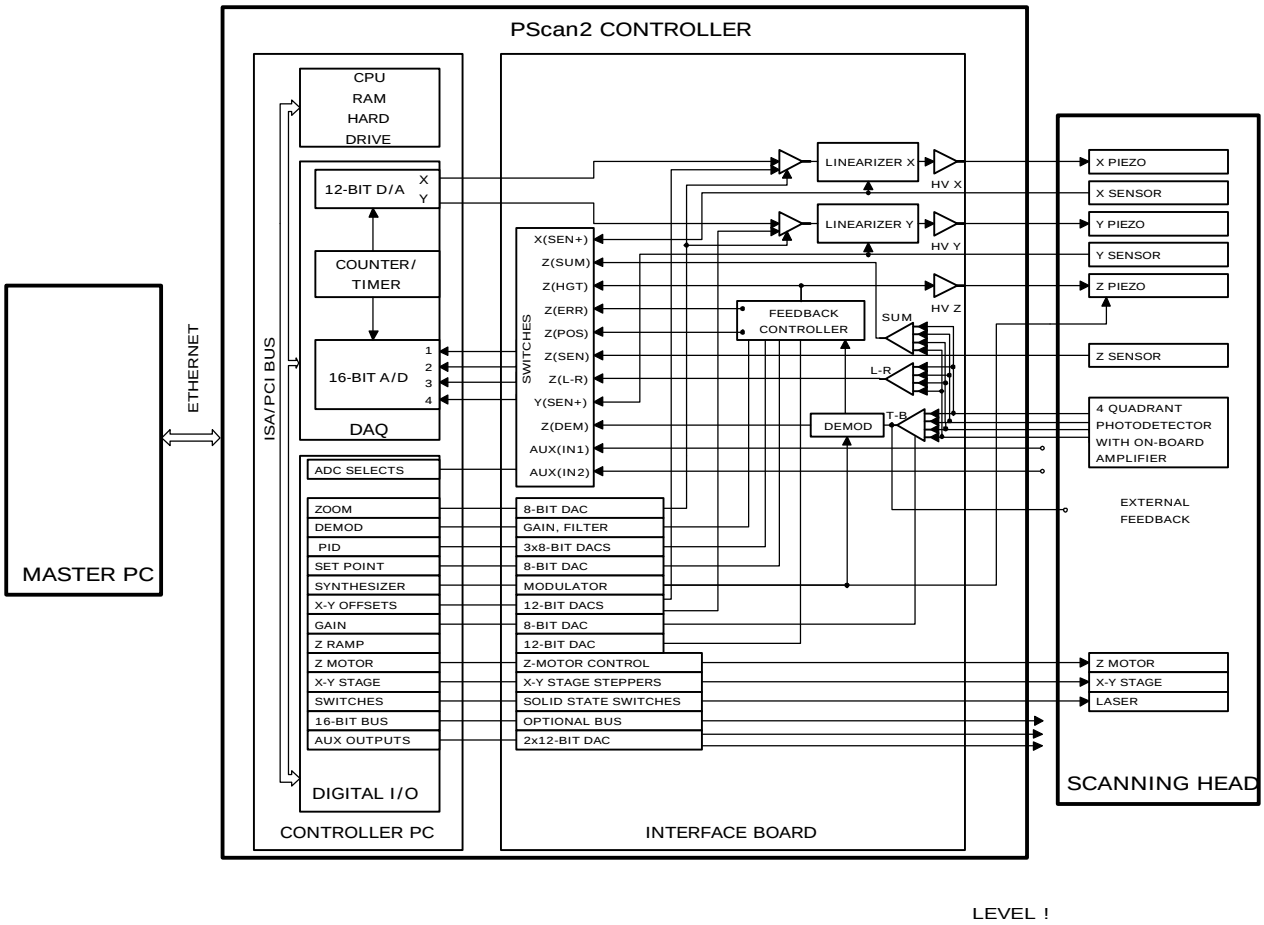
POWER IN, HIGH VOLTAGE

CONNECTOR # 8: 6 PIN 0.156 WALDOM HEADER, MALE

1	GND	BLK	**4**	+ 140 V	VIOL
2	NC	***	**5**	NC	***
3	NC	***	**6**	NC	***

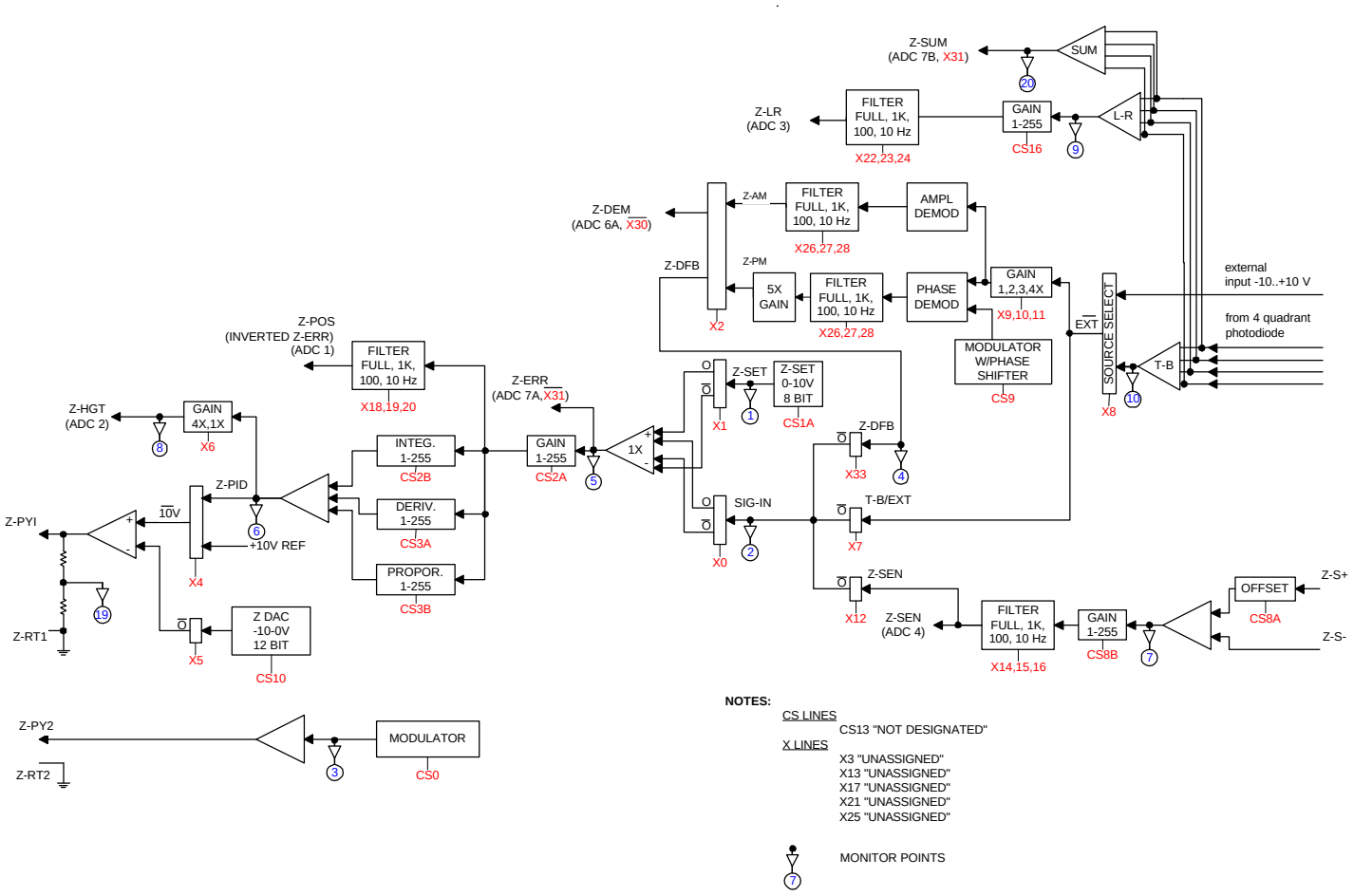
Appendix C: Block Diagrams for PScan2™ Controller (Level 1 and Level 2)

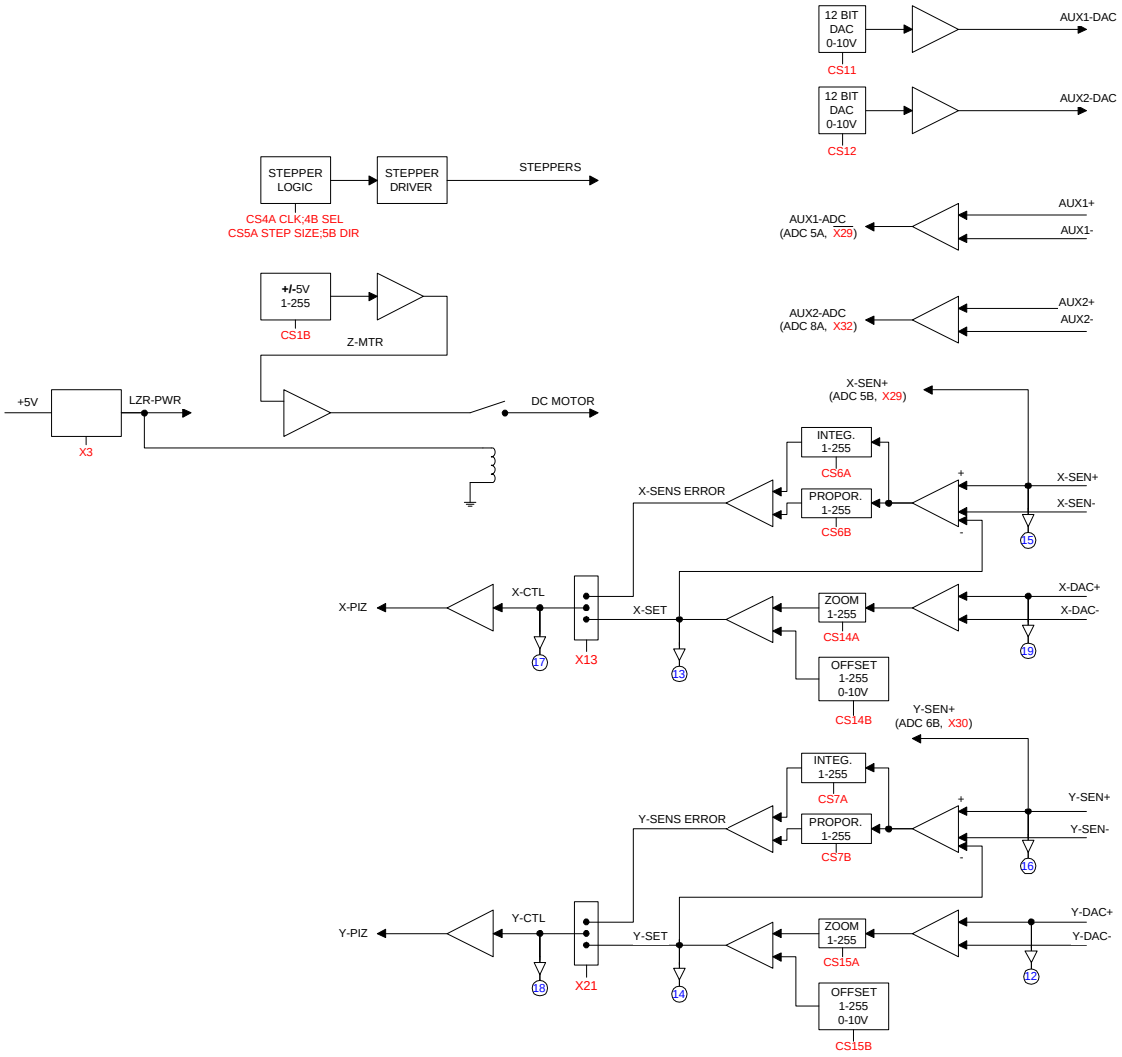
Master-Slave Electronics Block Diagram



LEVEL !

Appendix D: Schematic Diagrams for PScan2™ Controller Rev. B




12 BIT DAC 0-10V
CS11
AUX1-DAC
12 BIT DAC 0-10V
CS12
AUX2-DAC
STEPPER LOGIC
STEPPER DRIVER
STEPPERS
CS4A CLK;4B SEL
CS5A STEP SIZE;5B DIR
AUX1-ADC (ADC 5A, X29)
AUX1+
AUX1-
+/-5V 1-255
CS1B
Z-MTR
AUX2-ADC (ADC 8A, X32)
AUX2+
AUX2-
+5V
LZR-PWR
X3
DC MOTOR
X-SEN+ (ADC 5B, X29)
INTEG. 1-255
CS6A
X-SENS ERROR
PROPOR. 1-255
CS6B
X-SEN+
X-SEN-
15
X-PIZ
X-CTL
17
X13
X-SET
13
ZOOM 1-255
CS14A
X-DAC+
X-DAC-
19
OFFSET 1-255 0-10V
CS14B
Y-SEN+ (ADC 6B, X30)
INTEG. 1-255
CS7A
Y-SENS ERROR
PROPOR. 1-255
CS7B
Y-SEN+
Y-SEN-
16
Y-PIZ
Y-CTL
18
X21
Y-SET
14
ZOOM 1-255
CS15A
Y-DAC+
Y-DAC-
12
OFFSET 1-255 0-10V
CS15B

Appendix E: PScan2™ Controller Network Configuration

The Controller network configuration is stored in a file **"drives.bat"** on the Controller's hard disk drive. Example contents of this file is shown below:

```
net use I: \\Master\PScan2™ password /PERSISTENT:NO /YES
```

where **"net use"** is a Microsoft Network Client for DOS ver. 3.0 command, **"I:"** is a network drive letter (drive I: is used by default by the Controller software), **"\\Master"** is a Master Workstation=s network name (see Figure 11), **"\PScan2™"** is a name of shared resource on the Master Workstation B Device Directory, **"/PERSISTENT:NO"** is a command switch, and **"/YES"** means positive answer to all command questions.

A complete reference on **"net use"** command usage is given below:

```
Connects or disconnects your computer from a shared
resource or displays information about your connections.

NET USE [drive: |*] [\\computer\directory [password |?]]
    [/PERSISTENT:YES | NO] [/SAVEPW:NO] [/YES] [/NO]
NET USE [port:] [\\computer\printer [password | ?]]
    [/PERSISTENT:YES | NO] [/SAVEPW:NO] [/YES] [/NO]
NET USE drive: | \\computer\directory /DELETE [/YES]
NET USE port: | \\computer\printer /DELETE [/YES]
NET USE * /DELETE [/YES]
NET USE /PERSISTENT:YES|NO|LIST|SAVE|CLEAR [/YES] [/NO]
NET USE drive: | * /HOME
```

drive	Specifies the drive letter you assign to a shared directory.
*	Specifies the next available drive letter. If used with /DELETE, specifies to disconnect all of your connections.
port	Specifies the parallel (LPT) port name you assign to a shared printer.
computer	Specifies the name of the computer sharing the resource.
directory	Specifies the name of the shared directory.
printer	Specifies the name of the shared printer.
password	Specifies the password for the shared resource.
?	Specifies that you want to be prompted for the password of the shared resource. You don't need to use this option unless the password is optional.
/PERSISTENT	Specifies which connections should be restored the next time you log on to the network. It must be followed by one of the values below: YES Specifies that the connection you are making and any subsequent connections should be persistent. NO Specifies that the connection you are making and any subsequent connections should not be persistent. LIST Lists your persistent connections. SAVE Specifies that all current connections should be persistent. CLEAR Clears your persistent connections.
/SAVEPW:NO	Specifies that the password you type should not be saved in your password-list file. You need to retype the password the next time you connect to this resource.
/YES	Carries out the NET USE command without first prompting you to provide information or confirm actions.
/DELETE	Breaks the specified connection to a shared resource.
/NO	Carries out the NET USE command, responding with NO automatically when you are prompted to confirm actions.
/HOME	Makes a connection to your HOME directory if one is specified in your LAN Manager or Windows NT user account.

To list all of your connections, type NET USE without options.

To see this information one screen at a time, type the following at the command prompt:

```
NET USE /? | MORE
- or -
NET HELP USE | MORE
```

Appendix F: DCEx™ Protocol Components

1. PSCAN2™ SCANNING PROBE MICROSCOPE SYSTEM CONFIGURATION

1.1. SYSTEM COMPONENTS

PScan2™ SPM System Consists Of:

- A Master Workstation that operates under MS Windows 95, 98, NT, or XP™ operating systems and runs Application Software;
- A Controller that operates under MS-DOS 6.22™ and runs Controller Software;
- A Scanner that is connected to Controller's Interface Board;
- An Ethernet network link between the Master Workstation and Controller that is implemented via a Twisted Pair (TP) DirectLink Ethernet cable, two regular TP cables and Ethernet TP-Hub, or a coaxial Ethernet cable and two Ethernet network cards (10Mbps or 100Mbps) in the Master Workstation and Controller correspondingly. (See Figure 1 for basic system block diagram).

1.2. NETWORK SOFTWARE COMPONENTS

The current configuration operates using Microsoft NetBIOS Extended User Interface (NetBEUI) protocol on both Master Workstation and Controller.

Microsoft Network Client version 3.0 for MS-DOS is installed on the Controller side and provides a file-level network access to the Master Workstation's shared resources.

The following Network software components are required on the Master Workstation:

- Ethernet Adapter driver (Figure 2)
- NetBEUI protocol driver (Figure 4)
- Client for Microsoft Networks (Figure 6)
- File and Printer Sharing for Microsoft Networks service (Figure 7)

(Tip: **Use Settings->ControlPanel->Network** to add required network components or to edit their properties).

The necessary protocol bindings are required on Master Workstation:

- NetBEUI to Ethernet Adapter (Figure 3)
- Client and File&Printer Sharing to NetBEUI (Figure 5)

The File Sharing capability on the Master Workstation must be enabled (**Settings-> Control Panel-> Network-> File and Print Sharing->** "I want to be able to give others access to my files" checkbox checked - see Figure 8).

1.3. SHARED COMMUNICATION SPACE–DEVICE DIRECTORY

The **DCEx™™** protocol is a file-level protocol; that is, all commands, messages and data are represented by file structures. The communication space that hosts all these file structures constitutes a shared directory on the Master Workstation's hard disk drive called *Device Directory*. This directory has to be shared with access type **"full"** and can be password protected (see Figures 9, 10).

The Controller maps its network drive to the Device Directory and the Master Workstation's network name at a boot time. Therefore, before the Controller is switched on, the Master Workstation must be up and running, all required network software components installed and the Device Directory shared. The Master Workstation network name is set in **Settings->Control Panel->Network-> Identification.** The Controller's network configuration is described in Appendix1.

2. DATA COMMAND EXCHANGE (DCEX™™) PROTOCOL STRUCTURE

The Data Command Exchange (**DCEx™™**) protocol is an *Application level* protocol thatcan be used to send a command to the Controller, receive data from the Controller, get a message or status information from the Controller, or supply configuration parameter values to the Controller. There are four major groups of **DCXE™** protocol components – *Command files, Data files, Log files and one Configuration file* (Table 1). These are described below.

Commands constitute empty files (except CHANGE_FLAG) that are created by the Master Workstation and checked/deleted by the Controller. They are used to put the Controller into one of the designated functional modes, exit from a current mode (STOP_FLAG) or notify the Controller about configuration parameter value changes (CHANGE_FLAG). The CHANGE_FLAG file contains a text line with the parameter's section name that was changed and needs to be reapplied. If this file contains more than one line, then the Controller services only the last line entry.

Important note: In order to exit from a current functional mode, the Controller needs a STOP_FLAG command. This is because mode commands cannot interrupt each other (except DCMTR_FWD and DCMTR_REV mode commands which allow for an interrupt).

Data files are created and filled by the Controller and can be read by the Application software on the Master Workstation. These Data files contain ADC measurements for oscilloscope modes, for the frequency sweep mode, for scanning mode and Red Dot alignment mode. They are stored in either binary or ASCII format, depending on volume and throughput.

The **Log** file (ERROR.LOG) is created and filled by the Controller and can be accessed by the Application software on the Master Workstation. It is a text file in which each line is a message line or status line from the Controller. The Controller sends an empty line when it comes to the Idle mode. The last line of the ERROR.LOG represents the most recent message from the Controller.

The **Log** file (LINE.LOG) contains one text line with the number of scan lines for which data has already been acquired. It can be used by the Application software for scan progress monitoring and scan image data tracking.

The **Configuration** file SLAVE.INI is represented as a generic INI-file structure:

```
[SECTION1 NAME]
KEY1_NAME=KEY1_VALUE
KEY2_NAME=KEY2_VALUE
….
[SECTION2 NAME]
KEY3_NAME=KEY3_VALUE
KEY4_NAME=KEY4_VALUE
….
```

The section name must be in square brackets. The parameter description line starts with a key name, followed by "=" sign, then by the parameter value and ends with an "Enter". The order of keys within a section and the order of sections are not important. The detailed description of all configuration parameters, their values and related Controller hardware signals is provided in separate documents: Slaveini.xls and Slaveini.doc.

3. A TYPICAL EXAMPLE OF A DCEX™™ COMMUNICATION TRANSACTION

Let's assume that the Controller is in "Oscilloscope time"– mode and the next activity is a "Scan"-mode operation. Then the typical Application software actions would include the following:

- Issue a STOP_FLAG command to exit from the current mode (creates file "stop.flg" in Device Directory);
- Modify parameters in SLAVE_INI file, if needed;
- Issue a SCAN_START command to start scan operation (creates file "scanstrt.flg" in Device Directory);
- Periodically, read text line from the LINE_LOG file and read the corresponding data set from the SCAN_DAT file, and update scan progress indicator and process/display image;
- Modify parameters in the SLAVE_INI file and create a CHANGE_FLAG file in the Device Directory with a changed section name in it (one at a time), if needed, while scan operation is still in progress.
- Issue a STOP_FLAG command, if needed, in order to terminate scan operation before its completion.

4. COMMANDS AND CONTROLLER'S FUNCTIONAL MODES

The Controller is designed to operate in one of the specific functional modes. There are currently 13 functional modes. Each functional mode represents a specific task performed by the Controller. Functional mode can be either time-unlimited or time-limited. An example of a time-unlimited mode is the "Oscilloscope, time mode". The Controller is allowed to stay in this mode as long as appropriate. An example of a time-limited mode is the "Scan Image" mode. The Controller will exit this mode as soon as the scan operation is completed.

DCEx™™ commands are used to navigate the Controller through functional modes, initiate a specific operation or abort current operation (STOP_FLAG), request current status (PING_FLAG) or notify the Controller about operating parameters change (CHANGE_FILE).

In addition to 13 functional modes there are two "Standalone" modes that are designed for the Controller's network configuration and the Controller's software update. "Standalone" here means that the Controller is not connected to the Master workstation. There are two "Standalone" commands, CONFIGURE_FLAG and

UPDATE_FLAG. “Standalone” commands are supplied to the Controller via floppy disk drive and are checked by the Controller only during boot up and only in standalone configuration (the Controller is not connected to the Master workstation).

4.1. IDLE MODE

```
Command: STOP_FLAG
Default mode for power on and reboot
Time-unlimited
Logs used: ERR_LOG, PING_LOG, PID_LOG
Data files: none
```

The Idle mode is the default mode that the Controller enters after first power up, reboot or after the STOP_FLAG command is issued. Being in this mode, the Controller polls Device Directory for the occurrence of any Command flag (command file). The following cycled order is used for commands polling:

- Reset (RESET_FLAG);
- Ping (PING_FLAG);
- Change (CHANGE_FILE);
- Tip retract (TIP_UP);
- Change (CHANGE_FILE);
- Tip approach (TIP_DOWN);
- Change (CHANGE_FILE);
- Red Dot alignment (REDDOT_START);
- Change (CHANGE_FILE);
- Scan Image (SCAN_START);
- Change (CHANGE_FILE);
- Oscilloscope, time mode (OSC1_START);
- Change (CHANGE_FILE);
- Oscilloscope, line mode (OSC2_START);

- Change (CHANGE_FILE);
- Frequency sweep (SWEEP_START);
- Change (CHANGE_FILE);
- Oscilloscope, storage mode (OSCSTO_START);
- Change (CHANGE_FILE);
- Stepper motor (STEPPER_START);
- Change (CHANGE_FILE);
- DC motor forward (DCMTR_FWD);
- Change (CHANGE_FILE);
- DC motor reverse (DCMTR_REV);

Once command flag is detected, the Controller performs an appropriate action or enters into one of the functional modes.

The CHANGE_FILE command flag is checked every time between two functional mode command flag checks. If CHANGE_FILE is detected, the parameter values from the appropriate section of the Slave.ini file are applied.

Every time the Controller completes or aborts current functional mode operation, it returns into the Idle mode and proceeds with the command polling according to the cycled order above. Let us assume as an example that the Controller has just completed scan image operation, then it will enter the Idle Mode and check for the presence of the "Oscilloscope, time mode" (OSC1_START) command, then the "Oscilloscope, line mode" (OSC2_START) command and so on. Let us assume further, that the Controller encounters the "Oscilloscope, line mode" (OSC2_START) command. Then the Controller would enter into the "Oscilloscope, line mode" functional mode and operate there until stop command (STOP_FLAG) is issued. When the stop command is issued, the Controller returns to the Idle mode and continues command polling with the "Frequency sweep" command checked next (see cycled polling order above).

Whenever the Controller is initialized (on power up or software reboot), it writes the Controller Software version information and the "Device Initialized" line into the log file (ERR_LOG) and enters into the Idle mode.

4.2. RESET MODE

```
Command: RESET_FLAG
Time-limited
Logs used: ERR_LOG, PID_LOG
Data files: none
```

The purpose of the reset mode is to reinitialize the Controller's Interface Board and to reopen the log file (ERR_LOG). **The Controller's computer is not reinitialized, reset, or affected by any means during this mode, nor is the Controller's software reloaded. Hardware power on reset must be used for a complete Controller system reinitialization**. This mode is designed to handle network communication failures in network link between the Controller and the Master Workstation. It is recommended that the Application Software on the Master Workstation issues "reset" command (RESET_FLAG) right after the "stop" command (STOP_FLAG) every time it is loaded.

Let us assume as an example that the Controller is in the "Image scan" mode and then suddenly the Master Workstation hangs. The Controller will then be stuck on a network I/O operation. After the Master Workstation reboots the Controller resumes a network operation and continues functioning. All information that designated to data and log files that were open by Controller before the Master Workstation was rebooted is going nowhere and is lost. The Controller remains in the same functional mode that it was in at the moment of the Master Workstation hang up. When the Application Software issues the "stop" command, the Controller is in the Idle mode, but log messages are still going nowhere. Then the "reset" command forces the Controller to reopen the log file and the Controller is ready to proceed with operation.

Whenever the Controller services the "reset" command, it writes the Controller Software version information and the "Device Ready" line into the log file (ERR_LOG).

4.3. TIP RETRACT MODE

```
Command: TIP_UP
Time-limited
Logs used: ERR_LOG, PID_LOG
Data files: none
```

This mode is used for SPM probe tip retract operation. During this operation the Controller first writes the "Tip Retract" line into the log file (ERR_LOG), then performs "fast retract" by activating the "fast retract" line (X5). The Controller further accesses parameter ZMTR_TIP in the Slave.ini file, section [TIP APPROACH], and uses its value for a Z-motor selection.

If Z DC motor is selected, the Controller accesses parameter DCREV_TIP and DCTIME_TIP values in the Slave.ini file, section [TIP APPROACH]. The values are used to apply a specified DC motor voltage for a specified period of time.

If one of the eight stepper motors is selected, the Controller accesses parameter DIRUP_TIP, STEPUP_TIP, PULSES_TIP and PACKET_TIP values in the Slave.ini file, section [TIP APPROACH]. The values are used to select the direction, full/half step, the number of pulses and pulse packet size for tip retraction using the stepper motor. Stepper pulses are produced at a 1 kHz rate, the network I/O operation (which takes additional time out of stepping) is performed only between pulse packets. Thus the PACKET_TIP value determines the actual speed of tip retraction using the stepper motor.

Tip retraction can be terminated before the completion (DCTIME_TIP elapsed time or PULSES_TIP stepper pulses) by a "stop" command. The Controller checks for a STOP_FLAG at about every 100 ms time interval if the DC motor is used and after each stepper pulse packet if the stepper motor is used. When the tip retraction is completed or terminated, the Controller writes an empty line into the log file (ERR_LOG).

4.4. TIP APPROACH MODE

```
Command: TIP_DOWN
Time-limited
Logs used: ERR_LOG, PID_LOG
Data files: none
```

This mode is used for SPM probe tip approach and Z-PID feedback engage operation. When the Controller enters into this mode it writes the "Tip Engage" line into the log file (ERR_LOG). Then the Controller sets the Z PID On/Off switch (X4) into the state according to the value of the PID_ON parameter (Slave.ini file, [PID ON/OFF] section). The Controller further sets the Z-DAC output to 0Volt level that means Z piezo is fully extended. After that the Controller accesses parameters ZMTR_TIP, CH_TIP and SRF_TIP from the Slave.ini file, section [TIP APPROACH].

If the Z DC motor is selected, the Controller accesses parameter DCFWD_TIP in the Slave.ini file, section [TIP APPROACH], and uses its value for Z DC motor DAC output. Then the Controller enters into the following loop:

- Check for a STOP_FLAG; if found, then activate fast retract line (X5), set Z DC motor DAC to zero output level, set Z DAC output to +10 Volt level (Z piezo fully retracted), deactivate fast retract line (X5) and terminate current mode;

- Acquire channel set by CH_TIP parameter value and compare acquired value with the SRF_TIP value; if value is close, then set Z DC motor DAC to zero output level, activate the Z PID On/Off switch (X4 line into ON state) and complete current mode.

If one of the eight stepper motors is selected by ZMTR_TIP parameter value, the Controller accesses parameters DIRDWN_TIP, STEPDWN_TIP and CYCLES_TIP in the Slave.ini file, section [TIP APPROACH], and uses their values for stepper direction, full/half step and acquisition rate selection. Then the Controller enters into the following loop:

- Generate one pulse for the selected stepper motor;

- Check for a STOP_FLAG; if found, then activate fast retract line (X5), set ZDAC output to +10 Volt level (Z piezo fully retracted), deactivate fast retract line (X5) and terminate current mode;

- Acquire channel set by CH_TIP parameter value and compare acquired value with the SRF_TIP value; if value is close, then activate the Z PID On/Off switch (X4 line into ON state) and complete current mode. Repeat current step the number of CYCLES_TIP value times.

Every acquisition cycle takes approximately 15 microseconds. The CYCLES_TIP value determines the number of acquisition cycles between step pulses. Thus the CYCLES_TIP value determines the actual speed of the tip approach using stepper motor.

When the “tip approach” mode is completed or terminated, the Controller writes an empty line into the log file (ERR_LOG).

4.5. RED DOT ALIGNMENT MODE

```
Command: REDDOT_START
Time-unlimited
Logs used: ERR_LOG, PING_LOG, PID_LOG
Data files: REDDOT_DAT
```

The "Red Dot Alignment" mode is designated to trace the position of a reflected laser beam on a four-quadrant photo-detector (AFM application). When the Controller enters into this mode, it first writes the "Red Dot alignment" line into the log file (ERR_LOG). Then the Controller applies parameter LR_G, LR_OFS, LR_F values from [INPUT SELECTS] section and parameter PID_POL, PID_SET, ZERR_G, Z_SET values from [Z FEEDBACK] section of the Slave.ini file. The Controller further selects T-B photo-detector signal as an input for Z feedback channel and selects to bypass the demodulator. Then the Controller enters into the following loop:

- Acquire Z_ERR, Z_LR, Z_SUM ADC input channel;
- Write acquired values into the data file REDDOT_DAT starting from its zero position, data represented as an ASCII text line (comma separated);
- Check for a STOP_FLAG; if found, then output an empty line into the log file (ERR_LOG) and terminate current mode;
- Check for a CHANGE_FLAG; if found, then apply parameter values from an appropriate section of the Slave.ini file.

When the "Red Dot Alignment" mode is terminated, the Controller writes an empty line into the log file (ERR_LOG).

4.6. SCAN IMAGE MODE

```
Command: SCAN_START
Time-limited
Logs used: ERR_LOG, LINE_LOG, PING_LOG, PID_LOG
Data files: SCAN_DAT (aka OSC2_DAT)
```

This mode is designed for SPM image acquisition. When the Controller enters into this mode, it first writes the "Scan Image" line into the error log file (ERR_LOG). Then the Controller opens the line log file (LINE_LOG) and outputs "0" line into it, which means no line is scanned at that moment. The Controller further accesses parameter values in the Slave.ini file, section [SCAN IMAGE], which are used for Scan Image operation. Before the actual Scan Image operation is started, the Controller applies parameter values for the following sections of the Slave.ini file: [INPUT SELECTS], [XY CONTROL], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [AUX 1&2], [LASER]. Then the Controller carries out the "slow" tip position initialization, the SPM tip is moved from its current arbitrary XY position to the scan start XY point. The tip is moved via a straight line using the number of POINTS increment with the rate of a given SCAN_RATE.

The actual Scan Image operation consists of the number of LINES alternating "forward" line scan and "reverse" line scan operations. The acquired data are transferred into the data file (SCAN_DAT) after line scan operation depending on the acquisition direction parameter DIR value. If DIR value is 0 ("forward" scan), then data are transferred only after "forward" line scan operations. If DIR value is 1 ("reverse" scan), then data are transferred only after "reverse" line scan operations. And finally, if DIR value is 2 ("forward/reverse" scan), then data are transferred after both "forward" and "reverse" line scan operations. Whenever scan line data are transferred into the data file (SCAN_DAT), the Controller increments the scan line counter and writes its value into the line log file (LINE_LOG) starting from the zero file position. Thus the ASCII text line in the line log file always represents the number of the line scan data sets in the data file (SCAN_DAT).

The Controller checks for a stop flag (STOP_FLAG) after **each** line scan operation. If the stop flag is found, the Controller writes an empty line into the log file (ERR_LOG) and terminates Scan Image operation.

The Controller also checks for a change flag (CHANGE_FILE) indicator after each line scan. If this flag is found and indicates that [INPUT SELECTS], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [PID ON/OFF], [Z PIEZO], [AUX 1&2] or [LASER] section was modified, then the Controller applies parameter values from that section of the Slave.ini file and continues Scan Image operation. Else if the change flag indicates [SCAN IMAGE] or [XY CONTROL]

modified section, the Controller writes the “Scan Image mode restarted” line into the log file (ERR_LOG) and restarts the Scan Image operation from the very beginning.

When Scan Image operation is completed, the Controller writes an empty line into the log file (ERR_LOG) and returns into the Idle mode.

4.7. OSCILLOSCOPE, TIME MODE

```
Command: OSC1_START
Time-unlimited
Logs used: ERR_LOG, PING_LOG, PID_LOG
Data files: OSC1_DAT
```

This mode is designed for 4 input channel acquisition at the real time scale. One hundred data point per channel is acquired during every TimeBase interval. Thus the time interval between two data points is equal to the TimeBase / 100. The TimeBase value can vary from 10 ms to 1000 ms. The acquired 100 point data are transferred as a whole set between every two TimeBase intervals, the time required for data transfer being lost from data acquisition.

When the Controller enters into this mode, it first writes the “Oscilloscope, time mode” line into the log file (ERR_LOG). Before the actual data acquisition is started, the Controller applies parameter values for the following sections of the Slave.ini file: [INPUT SELECTS], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [AUX 1&2]. The Controller further accesses parameter TIME_BASE value in Slave.ini file, section [OSC TIME], and uses this value as a TimeBase interval. After 100 data point are collected the Controller writes 100 16-bit values into the data file OSC1_DAT using binary format and starts next 100 data point acquisition. The Controller stays in the “Oscilloscope, time mode” until this mode is interrupted by a STOP_FLAG command.

It is permissible to change the TIME_BASE value during the “Oscilloscope, time mode” operation. The CHANGE_FILE command must be issued to force the Controller to apply an updated TIME_BASE value.

The Controller checks for a change flag (CHANGE_FILE) indicator after each series of data point acquisition. If this flag is found and indicates that [INPUT SELECTS], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [PID ON/OFF], [Z PIEZO], [AUX 1&2], [LASER] or [OSC TIME] section was modified, then the Controller applies parameter values from that section of the Slave.ini file and continues “Oscilloscope, time mode” operation.

4.8. OSCILLOSCOPE, LINE SCAN MODE

```
Command: OSC2_START
Time-unlimited
Logs used: ERR_LOG, LINE_LOG, PING_LOG, PID_LOG
Data files: OSC2_DAT (aka SCAN_DAT)
```

This mode is designed for repetitive acquisition of up to 4 selected input channels during one line of XY raster scanning. This mode is analogous to the Scan Image mode, except only one line is scanned. Data can be acquired during either forward or reverse or both directions of line scan.

When the Controller enters into this mode, it first writes the "Oscilloscope, line mode" line into the log file (ERR_LOG). Then the Controller opens the line log file (LINE_LOG) and writes the "0" line into it, which means no line is scanned at that moment. The Controller then accesses parameter values in Slave.ini file, section [SCAN IMAGE], which are used for line scan operation. Before the actual image scan operation is started, the Controller applies parameter values for the following sections of the Slave.ini file: [INPUT SELECTS], [XY CONTROL], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [AUX 1&2], [LASER]. Then the Controller carries out the "slow" tip position initialization. The SPM tip is moved from its current arbitrary XY position to the scan start XY point. The tip is moved via a straight line using the number of POINTS increment with the rate of a given SCAN_RATE.

The actual line scan operation consists of alternating "forward" line scan and "reverse" line scan operations. The acquired data are transferred into the data file (OSC2_DAT) after each line scan operation depending on the acquisition direction parameter DIR value. If DIR value is 0 ("forward" scan), then data are transferred only after "forward" line scan operations. If DIR value is 1 ("reverse" scan), then data are transferred only after "reverse" line scan operations. And finally, if DIR value is 2 ("forward/reverse" scan), then data are transferred after both "forward" and "reverse" line scan operations. Whenever scan line data are transferred into the data file (OSC2_DAT), the Controller increments the scan line counter and writes its value into the line log file (LINE_LOG) starting from the zero file position. Thus the ASCII text line in the line log file always represents the number of the line scan data sets in the data file (OSC2_DAT). This number can be 0 (no data currently available), 1 (data for one line scan are collected) or 2 (data for both "forward" and "reverse" lines are collected, DIR=2). The repetitive data for each line scan operation are written to the data file (OSC2_DAT) always starting from the zero file position.

The Controller checks for a stop flag (STOP_FLAG) after each line scan operation. If the stop flag is found, the Controller writes an empty line into the log file (ERR_LOG) and terminates line scan operation and returns into the Idle mode.

The Controller also checks for a change flag (CHANGE_FILE) indicator after each line scan. If this flag is found and indicates that [INPUT SELECTS], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [PID ON/OFF], [Z PIEZO], [AUX 1&2] or [LASER] section was modified, then the Controller applies parameter values from that section of the Slave.ini file and continues the "Oscilloscope, line scan" operation. Else if the change flag indicates [SCAN IMAGE] or [XY CONTROL] modified section, the Controller writes the "Line scan mode restarted" line into the log file (ERR_LOG) and restarts the "Oscilloscope, line mode" operation from the very beginning.

4.9. FREQUENCY SWEEP MODE

```
Command: SWEEP_START
Time-limited
Logs used: ERR_LOG, PING_LOG, PID_LOG
Data files: SWEEP_DAT
```

This mode is designed for a 4 input channel acquisition during frequency sweep on a numerically controlled oscillator. This mode allows an acquisition of a signal frequency response in a selected frequency range.

When the Controller enters into this mode, it writes the "Oscilloscope, frequency sweep mode" line into the log file (ERR_LOG). The Controller then accesses parameter values in the Slave.ini file, section [FREQ SWEEP], which are used for a frequency sweep operation. Before the actual frequency sweep operation is started, the Controller applies parameter values for the following sections of the Slave.ini file: [INPUT SELECTS], [XY CONTROL], [Z FEEDBACK], [DEMOD SELECTS], [AUX 1&2], [LASER]. During the actual frequency sweep operation the Controller programs the numerically controlled oscillator for 400 different frequency values evenly distributed over the frequency range determined by the FREQ_S and the FREQ_E values from the Slave.ini file, section [FREQ SWEEP]. The 400 frequency points are produced at the rate of approximately 12 ms, the overall frequency sweep duration is about 5-6 seconds. The data for 4 selected input channels are acquired for every frequency point on a "first acquire then increment" principle. Thus the settling time for every frequency point is approximately 12 ms. The values acquired for each frequency point are written by the Controller into the data file (SWEEP_DAT) using ASCII text format. Thus every line in the data file (SWEEP_DAT) contains four decimal values in ASCII text format representing ADC data for 4 input channels.

The Controller checks for a stop flag (STOP_FLAG) after each frequency point acquisition. If the stop flag is detected, the Controller writes an empty line into the log file (ERR_LOG), terminates the "Oscilloscope, frequency sweep mode" operation and returns into the Idle mode.

The Controller also checks for a change flag (CHANGE_FILE) indicator after each frequency point acquisition. If this flag is detected and indicates that [PID ON/OFF], [Z PIEZO], [AUX 1&2] or [LASER] section was modified, then the Controller applies parameter values from that section of the Slave.ini file and continues the "Oscilloscope, frequency sweep mode" operation. The Controller do not take any actions if the change flag indicates [INPUT SELECTS], [FREQ SWEEP], [Z FEEDBACK], [DEMOD SELECTS] or [XY CONTROL] modified section. The "Oscilloscope, frequency sweep mode" must be restarted by the user in order for the changes in sections mentioned above to take effect.

4.10. OSCILLOSCOPE STORAGE MODE

```
Command: OSCSTO_START, OSCSTO_NEXT
Time-unlimited
Logs used: ERR_LOG, PING_LOG, PID_LOG
Data files: OSCSTO_DAT
```

This mode is designed for 4 input channel acquisition at the real time scale. It is analogous to the "Oscilloscope, time mode" except longer TimeBase values are used. The name "Storage" is derived from an analogy to an electronic digital storage oscilloscope. As in the case of an electronic storage scope the "Oscilloscope storage mode" is useful for an acquisition of a "slow-changing" signal. Three hundred data points per channel are acquired during every TimeBase interval. Thus the time interval between two data points is equal to the TimeBase / 300. The TimeBase value can vary from 2000 ms to 10000 ms (2s to 10 s). The acquired data point values are transferred before the next data point is acquired. This transfer on a per-point basis allows an application on a Master Workstation to trace the data during the prolonged TimeBase interval, which may constitute from 2 to 10 seconds.

When the Controller enters into the given mode, it first writes the "Oscilloscope Storage mode" line into the log file (ERR_LOG). Before the actual data acquisition is started, the Controller applies parameter values for the following sections of the Slave.ini file: [INPUT SELECTS], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [AUX 1&2], [XY CONTROL]. The Controller further accesses parameter TIME_BASE and DUTY_TIME values in the Slave.ini file, section [OSC STORAGE]. The DUTY_TIME value is subtracted from the TIME_BASE value; the result is used by the Controller as a TimeBase value. The

DUTY_TIME value is designated for the calibration of an “Oscilloscope storage mode”. The idea is that the Controller spend some amount of time for an acquisition and data transfer and some correction of a delay between every two data points is required. The DUTY_TIME value may vary from 0 to 1900 ms.

After all 300 data points are collected, the Controller waits for an OSCSTO_NEXT command before proceeding with the next 300 data point acquisition. This “hand shake” confirmation allows the synchronization of the display procedure on the Master Workstation with the data acquisition procedure on the Controller. The Controller stays in the “Oscilloscope storage mode” until this mode is interrupted by a STOP_FLAG command.

It is permissible to change the TIME_BASE value during the “Oscilloscope storage mode” operation. The CHANGE_FILE command must be issued to force the Controller to apply an updated TIME_BASE value.

The Controller checks for a change flag (CHANGE_FILE) indicator after **each** data point acquisition. If this flag is detected and indicates that section [INPUT SELECTS], [Z FEEDBACK], [DEMOD SELECTS], [FREQUENCY SYNTH], [PID ON/OFF], [Z PIEZO], [AUX 1&2], [LASER], [XY CONTROL] or [OSC TIME] was modified, then the Controller applies parameter values from that section of the Slave.ini file and continues “Oscilloscope storage mode” operation.

4.11. STEPPER MOTOR MODE

```
Command: STEPPER_START
Time-limited
Logs used: ERR_LOG
Data files: none
```

This mode is designed to operate one of eight available stepper motors. Stepping pulses drive stepper motors; only one stepper motor at a time can be active in current mode. Multiple stepper motors should be operated consecutively via multiple STEPPER_START commands.

When the Controller enters into the given mode, it writes the “Stepper motor” line into the log file (ERR_LOG). Then the Controller accesses the parameter MOTOR, STEP_DIR, STEP, PULSES and PACKET values from the Salve.ini file, section [STEPPERS]. The MOTOR value selects one of the eight stepper motors available, the STEP_DIR value selects either “forward” or “reverse” stepping direction, and the STEP value selects either “full” or “half” step. Parameter PULSES value defines the overall number of stepping pulses to be output to the stepper motor. Packets output

stepping pulses; the number of pulses per packet is defined by the PACKET value. The Controller performs network input/output operation only between packets; therefore the actual rotation speed of the stepper motor is defined by the PACKET value. The default value of PACKET is 1.

The Controller forms 1 ms duration stepping pulses and checks for the STOP_FLAG command every time between pulse packets. If STOP_FLAG is detected, the Controller aborts current mode operation and returns to the Idle mode.

When the Controller terminates or aborts the stepper motor mode it writes an empty line into the log file (ERR_LOG).

4.12. DC MOTOR FORWARD MODE

```
Command: DCMTR_FWD
Time-limited
Logs used: ERR_LOG
Data files: none
```

This mode is designed for the Direct Current (DC) motor operation. A control voltage drives the DC motor on a Digital to Analog Converter (DAC) that may vary from –5,000 mV to +5,000 mV. Different control voltage polarity yields to different DC motor rotation direction. Thus “forward” and “reverse” DC motor direction depends on a custom hardware wiring of DC motor. The two DC motor related modes of the Controller operation allows the user to define which control voltage is considered “forward” and which one is considered “reverse”.

When the Controller enters into the described mode, it writes the “DC Motor Forward” line into the log file (ERR_LOG). Then the Controller accesses the parameter DCMTR_TIME and DCMTR_FWD values from the Salve.ini file, section [DC MOTOR]. The DCMTR_TIME value specifies the duration of a DC motor action and should be a multiple of 100 ms. The DCMTR_FWD value specifies the control voltage that may vary from –5,000 mV to +5,000 mV. The polarity of the control voltage determines the direction of DC motor rotation; the amplitude determines the speed of DC motor rotation.

The Controller outputs the control voltage specified by the DCMTR_FWD value to the DC motor DAC and enters into the following cycle:

- Check for a STOP_FLAG command; if detected, abort current mode operation;

- Wait 100 ms and compare elapsed time with the DCMTR_TIME value; if equal, then terminate current mode operation.

When the Controller terminates or aborts the "DC Motor forward" mode operation, it sets DC motor DAC to a zero volt level and writes an empty line to the log file (ERR_LOG).

4.13. DC MOTOR REVERSE MODE

```
Command: DCMTR_REV
Time-limited
Logs used: ERR_LOG
Data files: none
```

This mode is designed for the Direct Current (DC) motor operation. A control voltage drives the DC motor on a Digital to Analog Converter (DAC) that may vary from –5,000 mV to +5,000 mV. Different control voltage polarity yields to different DC motor rotation direction. Thus "forward" and "reverse" DC motor direction depends on a custom hardware wiring of DC motor. The two DC motor related modes of the Controller operation allows the user to define which control voltage is considered "forward" and which one is considered "reverse".

When the Controller enters into the described mode, it writes the "DC Motor Reverse" line into the log file (ERR_LOG). Then the Controller accesses the parameter DCMTR_TIME and DCMTR_REV values from the Salve.ini file, section [DC MOTOR]. The DCMTR_TIME value specifies the duration of a DC motor action and should be a multiple of 100 ms. The DCMTR_REV value specifies the control voltage that may vary from –5,000 mV to +5,000 mV. The polarity of the control voltage determines the direction of DC motor rotation; the amplitude determines the speed of DC motor rotation.

The Controller outputs the control voltage specified by the DCMTR_REV value to the DC motor DAC and enters into the following cycle:

- Check for a STOP_FLAG command; if detected, abort current mode operation;

- Wait 100 ms and compare elapsed time with the DCMTR_TIME value; if equals, then terminate current mode operation.

When the Controller terminates or aborts the "DC Motor reverse" mode operation, it sets DC motor DAC to a zero volt level and writes an empty line to the log file (ERR_LOG).

4.14. AUTO-CONFIGURATION STANDALONE MODE

```
Command: CONFIGURE_FLAG
Standalone Controller
Logs used: none
Data files: none
```

The auto-configuration standalone mode is designed for the Controller's network configuration. "Standalone" here means that the Controller is not connected to the Master workstation. "Standalone" commands are supplied to the Controller via floppy disk drive and are checked by the Controller only during boot up and only in standalone configuration (the Controller is not connected to the Master workstation).

The Controller attempts to connect to the Master workstation specified by the "net use" command in the "drives.bat" file and create a log file (ERR_LOG) in the Device Directory every time the Controller is reboot. If the connect fails (Ethernet cable not connected, specified Master workstation name or Device Directory name do not exist in the network or access password is invalid), the Controller checks the floppy disk drive "A:\". If the floppy disk is present in the drive, the Controller first checks for the CONFIGURE_FLAG command represented by an empty "configur.flg" file. If this command is detected, the Controller enters into the auto-configuration mode. If no CONFIGURE_FLAG command is detected, the Controller checks for the UPDATE_FLAG command represented by an empty "update.flg" file. If an UPDATE_FLAG command is detected, the Controller enters into the auto-update mode.

The Controller in the auto-configuration mode copies the "drives.bat" file from the floppy disk to the Controller's hard disk drive. If operation is completed successfully, the Controller produces the sound indication of 4 short beeps and halts the system. In case of an error the Controller produces the sound indication of 1 long beep and halts the system. The error message is output to the Controller's console.

When the Controller completes the auto-configuration mode operation, it always halts the system. The Controller must be rebooted in order for the configuration changes to take effect.

4.15. AUTO-UPDATE STANDALONE MODE

```
Command: UPDATE_FLAG
Standalone Controller
Logs used: none
Data files: none
```

The auto-update standalone mode is designed for the Controller's software update. "Standalone" here means that the Controller is not connected to the Master workstation. "Standalone" commands are supplied to the Controller via floppy disk drive and are checked by the Controller only during boot up and only in standalone configuration (the Controller is not connected to the Master workstation).

The Controller attempts to connect to the Master workstation specified by the "net use" command in the "drives.bat" file and create a log file (ERR_LOG) in the Device Directory every time the Controller is reboot. If the connect fails (Ethernet cable not connected, specified Master workstation name or Device Directory name do not exist in the network or access password is invalid), the Controller checks the floppy disk drive "A:\". If the floppy disk is present in the drive, the Controller first checks for the CONFIGURE_FLAG command represented by an empty "configur.flg" file. If this command is detected, the Controller enters into the auto-configuration mode. If no CONFIGURE_FLAG command is detected, the Controller checks for the UPDATE_FLAG command represented by an empty "update.flg" file. If an UPDATE_FLAG command is detected, the Controller enters into the auto-update mode.

The Controller in the auto-update mode saves the current version of the Controller's software executable as a "pscan.bak" file and copies the "pscan.exe" file from the floppy disk to the Controller's hard disk drive. If operation is completed successfully, the Controller produces the sound indication of 6 short beeps and halts the system. In case of an error the Controller produces the sound indication of 1 long beep and halts the system. The error message is output to the Controller's console.

When the Controller completes the auto-update mode operation, it always halts the system. Reboot the Controller for the software update to take effect.

Appendix G: “slave.ini” File Structure including Sample File

PSCAN2™ CONFIGURATION FILE: [SLAVE.INI]

```
[INPUT SELECTS]  // input selects to ADC section
CH1=0            // Channel 1 input select:          0 – Z(POS)
                 //                                  1 – Z(HGT)
                 //                                  2 – Z(L-R)
                 //                                  3 – Z(SEN)
                 //                                  4 – AUX(IN1)
                 //                                  5 – X(SEN)
                 //                                  6 – Z(DEM)
                 //                                  7 – Y(SEN)
                 //                                  8 – Z(ERR)
                 //                                  9 – Z(SUM)
                 //                                  10 – AUX(IN2)
                 //                                  11 – ADC8B
CH2=1            //
CH3=2            // Channel 2 input select (0..11)
CH4=3            // Channel 3 input select (0..11)
ZSEN_G=255       // Channel 4 input select (0..11)
ZSEN_O=255       // Z sensor Gain (1..255)
ZSEN_F=0         // Z sensor Offset (0..255)
                 // Z sensor Filter:                 0 – Full range
                 //                                  1 – 1000 Hz
                 //                                  2 – 100 Hz
                 //                                  3 – 10 Hz
ZPOS_F=2         // error signal Z-POS filter:       0 – Full range
                 //                                  1 – 1000 Hz
                 //                                  2 – 100Hz
                 //                                  3 – 10Hz
ZHGT_G=0         // Z-HGT Gain select(0 – 1x, 1 – 4x)
```

```
                  // Lateral Force L-R Gain value (1..255)
                  // Lateral Force L-R Offset value (0..255)
                  // Lateral Force Filter:          0 – Full range
                  //                                1 – 1000 Hz
                  //                                2 – 100Hz
                  //                                3 – 10Hz
                  //

[Z FEEDBACK]      // Z feedback parameters section
PID_CH=0          // PID Channel select:       0 – T-B Photodetector
                  //                           1 – External
                  //                           2 – Z-SEN
PID_DEM=1         // PID Demodulator select:   0 – Bypass demodulator
                  //                           1 – Demodulate
PID_POL=0         // PID Input polarity select:
                  //                                0 – NORMAL
                  //                                1 - INVERSE
PID_SET=0         // Setpoint polarity select:
                  //                                0 – NORMAL
                  //                                1 - INVERSE
ZERR_G=255        // error signal Z-ERR gain select (1..255)
PID_P=255         // PID value 1 – proportional (0..255)
PID_I=255         // PID value 2 – integral (0..255)
PID_D=255         // PID value 3 – derivative (0..255)
Z_SET=0           // Z Setpoint value (0..255)(0..+10,000) mV

[PID ON/OFF]      // Z PID feedback On/Off section
PID_ON=0          // PID on/off select (0 – off, 1 – on)

[DEMOD SELECTS]   // Demod selects section
DEM_G=3           // Demod Gain (0 – 1x, 1 – 2x, 2 – 3x, 3 – 4x)
DEM_F=0           // Demod Filter:                  0 – Full range
                  //                                1 – 1000 Hz
                  //                                2 – 100Hz
                  //                                3 – 10Hz
DEMOD=0           // Demodulation mode (0 – phase, 1 – amplitude)
```

```
[Z PIEZO]            // Z piezo and sensor selects
                  // Z DAC output to drive piezo (0..4095)(0–10,000 mV)
                  // Fast Retract:      1 – Fully retracted
                  //                    0 – Z DAC applied to Z piezo
[XY CONTROL]         // X-Y Control Section
X_OFS=255         // X offset (0..255)(0..10,000 mV)
Y_OFS=255         // Y offset (0..255)(0..10,000 mV)
ZOOM=255          // X&Y Zoom (0..255)
XFBK_P=255        // X feedback proportional value (1..255)
XFBK_I=255        // X feedback integral value (0..255)
YFBK_P=255        // Y feedback proportional value (1..255)
YFBK_I=255        // Y feedback integral value (0..255)
XPI_ON=1          // X feedback On/Off state (0 – open loop, 1 – closed loop)
YPI_ON=1          // Y feedback On/Off state (0 – open loop, 1 – closed loop)
EXTRA_ZOOM=1      // Extra zoom on ACL DACs (1 – 1x default, 2 – 2x, 4 – 4x)
EXTRA_XOFS=1024   // Extra X Offset on ACL DAC (0..4095)
EXTRA_YOFS=1024   // Extra Y Offset on ACL DAC (0..4095)

[FREQUENCY SYNTH]     //Frequency Synthesizer Section
FREQ=4294967295   // Frequency select (0..4294967295)(20..1,000 kHz)
F_AMP=512         // Amplitude select (0..512) (0..10,000 mV)
PHASE=0           // Phase shift (0..4095) (0.00..360.00 deg.)

[AUX 1&2]            // AUX 1&2 Output selects Section
AUX1=4095         // AUX 1 output (0..4095)(0..10,000 mV)
AUX2=4095         // AUX 2 output (0..4095)(0..10,000 mV)

[STEPPERS]           //Steppers Section
MOTOR=4           // Stepper motor select (0..7)
STEP_DIR=1        // Stepper direction select(0 – Forward, 1 – Reverse)
STEP=1            // Stepper step select(0 – Full step, 1 – Half step)
PULSES=1000       // Pulses to output to stepper (0..65535)
PACKET=100        // Number of step pulses per packet, network IO between
```

```
                    // packets only. Controls speed of stepper motor. Default is
                    // 1 pulse per packet.

[LASER]                 // Laser control Section
LASER=1             // Laser on/off control ( 0 – off, 1 – on)

[DC MOTOR]              // Z DC Motor section (approach)
DCMTR_FWD=127       // DC motor forward voltage value(-128..127)(-5000..+5000)mV
DCMTR_REV=-128      // DC motor reverse voltage value(-128..127)(-5000..+5000)mV
DCMTR_TIME=1000     // DC motor "ON" time 100..65535 ms (by 100 ms increment)

[SCAN IMAGE]            // Scan Image Setup Section
POINTS=200          // Scan resolution (10..1500)
LINES=200           // Scan resolution – equals to POINTS
SCAN_RATE=20        // Scan rate, lines/s (0..1000)
ROTATE=-180         // Scan rotation (-360..+360) deg.
XYMODE= 1           // X-Y calculation mode select:   0 – on Master
                    //                                1 – on Slave
DIR=0               // line acquisition direction:    0 – Forward
                    //                                1 – Reverse
                    //                                2 – forward/reverse
CH=4                // Number of Channels to acquire during scan (1..4)
DATA=0              // Transfer scan data mode (0 – by Line, default; 1 – Whole)
SKEW=0.95           // Skew correction (-10.00..+10.00 deg.)
OVERSCAN=0          // Number of over-scan points (0..127, default - 0)
PRESCAN=0           // Number of pre-scan lines (0..127, default - 0)

[TIP APPROACH]          // Parameters for tip engage/retract mode
ZMTR_TIP= 0         // Select Z motor for tip retract/engage:   0 – Z DC motor
                    //                                       1..8 – Stepper motor
CH_TIP= 8           // Select input channel to monitor for "close to surface"
                    // condition (0..10), see [INPUT SELECTS] section
SRF_TIP=-32768      // Value of input parameter considered as "close to surface"
                    // (-32768..32767) (-10,000..10,000 mV)
DEV_TIP=328         // deviation from the "Surface" value(above)for tip approach
```

```
                    // (328..32767) (100..10,000 mV)
                    //
DCREV_TIP=-127      // Z DC Motor Reverse Voltage(-128..127)(-5000..5000 mV) for
                    // tip retract
DCTIME_TIP=10000    // Z DC Motor Reverse "ON" time  (100..65535)(ms) (by 100 ms
                    // increment)
DCFWD_TIP=127       // Z DC Motor Forward Voltage (-128..127)(-5000..5000 mV)for
                    // tip engage
DIRUP_TIP=0         // Stepper Direction (0 – Forward, 1 – Reverse) for tip
                    // retract
STEPUP_TIP=0        // Stepper Step (0 – Full, 1 – Half) for tip retract
PULSES_TIP=500      // Number of stepper step pulses for tip retract (0..65535)
DIRDWN_TIP=0        // Stepper Direction(0 – Forward, 1 – Reverse)for tip
                    // engage
STEPDWN_TIP=0       // Stepper Step (0 – Full, 1 – Half) for tip engage
CYCLES_TIP=500      // Number of acquisition cycles between approach steps
                    // (1..65535). Each acquisition is ~15us. Controls approach
                    // stepper speed. Default is 500. Used by Tip Approach only.
PACKET_TIP=100      // Number of step pulses per packet, network IO between
                    // packets only (1..65535).Controls retract speed of stepper
                    // motor. Default is 1 pulse per packet. Used by Tip Retract
                    // only.
TIP_INC=0           // 0 - Generic Tip Approach routine;
                    // 1 - Incremental Tip Approach routine
RUN_INC=20          // Number of stepper half-steps per each iteration of
                    // Incremental Tip Approach (default - 20 half-steps)
RAMP_INC=20         // Z-DAC ramp increment value (default - 20 or ~50mV)
DELAY_INC=1         // Delay (ms) per each Z-DAC ramp increment (default - 1 ms)
[FREQ SWEEP]            // Parameters for frequency sweep mode
FREQ_S=4294         // Start Frequency (32 bit) (0..20MHz)
FREQ_E=4294967      // End Frequency (32 bit) (0..20MHz)
F_AMP=512           // Modulation Amplitude(10 bit)(0..512)(0..10000 mV)
F_PHASE=0           // Phase shift (0..4095) (0.00..360.00 deg.)
SWEEP_RATE=10       // Sweep rate in ms per point (1..65,535 ms, 10 – default)
[OSC TIME]          // Parameters for oscilloscope time-mode
```

```
TIME_BASE=10      // Acquisition time period (10..1000 ms) – 100 points per
                  //TIME_BASE period are acquired

[OSC STORAGE]         // Parameters for storage scope mode
TIME_BASE=3000    // Acquisition time period (2000..10000 ms) – 300 points per
                      //TIME_BASE period are acquired
DUTY_TIME=1200    // Time used for scope calibration

[XYZ SCALE]       // X,Y,Z Scale sizes – not used by Controller's software,
                  // they are device specific and stored in Slave.ini.
X_SCALE=100.00    // 100.00 in X_UNITs (um) when fully zoomed out.
                  // X scale unit type:   0 – um, microns
                  //                      1 – nm, nanometers
                  //                      2 – A, angstroms
                  //                      3 – some arbitrary units
Y_SCALE=100.00    // 100.00 in Y_UNITs (um) when fully zoomed out
Y_UNIT=0          // Y scale unit type:   0 – um, microns
                  //                      1 – nm, nanometers
                  //                      2 – A, angstroms
                  //                      3 – some arbitrary units
Z_SCALE=10.00     // 10.00 in Z_UNITs (um) full scale (Z-HGT)
Z_UNIT=0          // Z scale unit type:   0 – um, microns
                  //                      1 – nm, nanometers
                  //                      2 – A, angstroms
                  //                      3 – some arbitrary units
                  //                      4 - mV, millivolts
ZS_SCALE=10000    // 10000 in Z_UNITs (nm) full scale (Z-SEN)
ZS_UNIT=1         // Z scale unit type:   0 – um, microns
                  //                      1 – nm, nanometers
                  //                      2 – A, angstroms
                  //                      3 – some arbitrary units
                  //                      4 - mV, millivolts
AUX1_SCALE=20000  // 20000 in AUX1_UNITs (mV) full scale (AUX-IN1)
AUX1_UNIT=4       // Z scale unit type:   0 – um, microns
```

```
                  //                      1 – nm, nanometers
                  //                      2 – A, angstroms
                  //                      3 – some arbitrary units
                  //                      4 - mV, millivolts
AUX2_SCALE=20000  // 20000 in AUX2_UNITs (mV) full scale (AUX-IN2)
AUX2_UNIT=4       // Z scale unit type:  0 – um, microns
                  //                      1 – nm, nanometers
                  //                      2 – A, angstroms
                  //                      3 – some arbitrary units
                  //                      4 - mV, millivolts
ADC_SCALE=20000   // 20000 in ADC_UNITs(mV) full scale(other ADC)
ADC_UNIT=4        // Z scale unit type:  0 – um, microns
                  //                      1 – nm, nanometers
                  //                      2 – A, angstroms
                  //                      3 – some arbitrary units
                  //                      4 - mV, millivolts

[TIP XY]          // Tip XY positioning mode(via XY-DACs, scan voltage)
X_POS=2000        // X-DAC output (12 bit, 0..4095, 0..10,000 mV)
Y_POS=1024        // Y-DAC output (12 bit, 0..4095, 0..10,000 mV)
PID_ON=1          // Turn PID On/Off (1/0) before moving
                  // 0 - PID Off(e.g. before Incremental Tip Approach);
                  // 1 - PID On (e.g. when on sample);

[IDLE PARK]       // Park Idle Piezos feature
ENABLE=1          // 0 - Disable, 1 - Enable
TIME=1            // Idle Timeout in minutes

[FORCE DIST]      // Force distance curve measurement
ZDAC_S=0          // Start value of Z-DAC output(12 bit,0..4095,0..10,000 mV)
ZDAC_E=4095       // End value of Z-DAC output (12 bit, 0..4095, 0..10,000 mV)
CH=8              // ADC signal channel for "deflection" measurement
FORCE_RATE=1      // Rate in milliseconds per point(overall 512 data points)
PIX=256           // Resolution in pixels per each curve
DEF_LIMIT=3276    // Deflection Limit (-10,000..+10,000 mV, -32767..+32767)
```

```
NUMBER=1          // Number of curves to acquire and average (1..100)
CONTINUE=0        // Continuous acquisition mode(0-disabled, 1-enabled)
F_SCALE=20.0      // Force calibration - full scale (corresponds to full ADC
                  // scale +/-10,000mV or -32768..+32767)
F_UNIT=nN         // Force calibration - unit name (characters)
SPRING_K=0.65     // Cantilever spring constant K in nN/nm (same as in N/m)

[NON LINEAR]      //Non-Linearity Correction
SCL_CORR=0        // Scale Non-linearity correction(0 - Disabled, 1 - Enabled)
SX1=0.956289878691135   // Xreal = Sx1*Xsen + Sx2*Xsen^2 + Sx3*Xsen^3 (Volts)
SX2=5.58554564856925E-03
SX3=8.97179016859237E-04
SY1=0.926743237609507   // Yreal = Sy1*Ysen + Sy2*Ysen^2 + Sy3*Ysen^3 (Volts)
SY2=4.23723511526169E-02
SY3=-6.48429909653124E-03
IMG_CORR=1        // Image Non-linearity correction(0-No,1-On-line,2-Off-line)
A1=1.04786333930482     // xsen = A1*xreal + A2*xreal^2 + A3*xreal^3 (pixels)
A2=-4.98515946291684E-04
A3=+5.04553013638955E-07
B1=1.04786333930482     // ysen = B1*yreal + B2*yreal^2 + B3*yreal^3 (pixels)
B2=-4.98515946291684E-04
B3=+5.04553013638955E-07
X_OFS0=69         // X offset of calibration image (0..255, 0..10,000 mV)
Y_OFS0=69         // Y offset of calibration image (0..255, 0..10,000 mV)
ZOOM0=120         // Zoom of calibration image (0..255)
PIX0=512          // Resolution of calibration image (128..1024 pixels)
[HYST CORR]  // Hysteresis correction model (Educational system, open loop XY)
ZOOM0=128    // The zoom of reference scans (must be forward at 180°)
PIX0=256     // The resolution of reference scans
OFS0=128     // The offset of lower-right reference scan (1st reference)
OPTB0=0.0008  // Hysteresis correction second order coefficient from FFT
OFS1=0       // The offset of upper-left reference scan (2nd reference)
OPTB1=0.0012  // Hysteresis correction' second order coefficient from FFT
PX0=1.36642  //X Scale Factor primary correction (vs. Zoom)
PX1=-0.00397     // Lx0 = px0 + px1*Zoom(LSB) + px2*Zoom(LSB)^2
```

```
PX2=0.0000102     //
PY0=1.23007       // Y Scale Factor primary correction (vs. Zoom)
PY1=-0.00331      // Ly0 = py0 + py1*Zoom(LSB) + py2*Zoom(LSB)^2
PY2=0.00000962    //
CX0=0.0000175     // X Scale Factor secondary correction (vs. X offset)
CX1=0.00000000233  // Lx = Lx0 + (cx0 + cx1*Xoffs(mV))*Xoffs(mV)
CY0=0.0000445     // Y Scale Factor secondary correction (vs. Y offset)
CY1=0             // Ly = Ly0 + (cy0 + cy1*Yoffs(mV))*Yoffs(mV)

A1=0.6215         // Hysteresis correction via FFT - Last known coefficients
A2=+0.001484      // Xreal = a1*Xpos + a2*Xpos^2 (pixels)
B1=0.6215         // Yreal = b1*Ypos + b2*Ypos^2 (pixels)
B2=+0.001484
[AUTO LINEARIZER]      // Auto-linearizer preferences
//!!! NO LONGER IN Master.ini SINCE VERSION 2.2.4!!!
//!!! MOVED INTO Slave.ini - DEVICE SPECIFIC!!!

XOFS_ADJUST=300  //X offset adjustment (-10,000..+10,000) mV
YOFS_ADJUST=300  //Y offset adjustment (-10,000..+10,000) mV
XPIX_ZOOM=0.9    //X pixel-zoom factor (0.00 .. 1.00)
YPIX_ZOOM=0.9    //Y pixel-zoom factor (0.00 .. 1.00)
```

Section	Parameter	Value	Control Lines	Comments	Value name
[INPUT SELECTS]	CH1	0	none	Select ADC0 on ACL	Z(POS)
		1	none	Select ADC1 on ACL	Z(HGT)
		2	none	Select ADC2 on ACL	Z(L-R)
		3	none	Select ADC3 on ACL	Z(SEN)
		4	X29=0	Select ADC4 on ACL	AUX(IN1)
		5	X29=1	Select ADC4 on ACL	X(SEN)
		6	X30=0	Select ADC5 on ACL	Z(DEM)
		7	X30=1	Select ADC5 on ACL	Y(SEN)
		8	X31=0	Select ADC6 on ACL	Z(ERR)
		9	X31=1	Select ADC6 on ACL	Z(SUM)
		10	X32=0	Select ADC7 on ACL	AUX(IN2)
		11	X32=1	Select ADC7 on ACL	ADC8B
	CH2	0..11	(see CH1 above)		
	CH3	0..11	(see CH1 above)		
	CH4	0..11	(see CH1 above)		
	ZSEN_G	1..255	CS8=0; D07..D00;A / B=1; WR=0	Z Sensor Gain DAC value	255..1
	ZSEN_O	0..255	CS8=0; D07..D00;A / B=0; WR=0	Z Sensor Offset DAC value	
	ZSEN_F	0	X16=1; X15=1; X14=1	Z Sensor Filter selection	Full Range
		1	X16=0; X15=1; X14=1		1000Hz
		2	X16=1; X15=0; X14=1		100Hz
		3	X16=1; X15=1; X14=0		10Hz
	ZPOS_F	0	X20=1; X19=1; X18=1	Z_POS Filter selection	Full Range
		1	X20=0; X19=1; X18=1		1000Hz
		2	X20=1; X19=0; X18=1		100Hz
		3	X20=1; X19=1; X18=0		10Hz
	ZHGT_G	0	X6=1	Z-HGT 1X Gain	1x
		1	X6=0	Z-HGT 4X Gain	4x
	LR_G	1..255	CS16=0; D07..D00;A / B=1; WR=0	L-R Gain DAC value	255..1
	LR_OFS	0..255	CS16=0; D07..D00;A / B=0; WR=0	L-R Offset DAC value	
	LR_F	0	X24=1; X23=1; X22=1	Lateral Force Filter selection	Full Range
		1	X24=0; X23=1; X22=1		1000Hz
		2	X24=1; X23=0; X22=1		100Hz
		3	X24=1; X23=1; X22=0		10Hz
[Z FEEDBACK]	PID_CH	0	X8=1; X12=1	T-B Photodet	
		1	X8=0; X12=1	External	
		2	X8 - Any; X12=0; X7=1; X33=1	Z-SEN (PID_DEM inactive)	
	PID_DEM	0	X7=0; X33=1	Bypass demod	
		1	X7=1; X33=0	Demod	
	PID_POL	0	X0=1	Normal polarity	NORMAL
		1	X0=0	Inverse polarity	INVERSE
	PID_SET	0	X1=0	Normal setpoint polarity	NORMAL

		1	X1=1	Inverse setpoint polarity	INVERSE
	ZERR_G	1..255	CS2=0; D07..D00;A / B=0; WR=0	Z-ERR Gain DAC value	255..1
	PID_P	0..255	CS3=0; D07..D00;A / B=1; WR=0	PID proportional DAC value	
	PID_I	0..255	CS2=0; D07..D00;A / B=1; WR=0	PID Integral DAC value	
	PID_D	0..255	CS3=0; D07..D00;A / B=0; WR=0	PID Derivative DAC value	
	Z_SET	0..255	CS1=0; D07..D00;A / B=0; WR=0	Z Setpoint DAC value (8 bit)	0..+10,000mV
[PID ON / OFF]	PID_ON	0	X4=0	PID off (Open Loop)	OFF
		1	X4=1	PID on	ON
[DEMOD SELECTS]	DEM_G	0	X9=1; X10=1; X11=1	Demodulator Gain selection	1x
		1	X9=0; X10=1; X11=1		2x
		2	X9=1; X10=0; X11=1		3x
		3	X9=1; X10=1; X11=0		4x
	DEM_F	0	X28=1; X27=1; X26=1	Demodulator Filter selection	Full Range
		1	X28=0; X27=1; X26=1		1000Hz
		2	X28=1; X27=0; X26=1		100Hz
		3	X28=1; X27=1; X26=0		10Hz
	DEMOD	0	X2=0	Demodulator mode selection	PHASE
		1	X2=1		AMPLITUDE
[Z PIEZO]	Z_OUT	0..4095	D11..D00; CS10=0 | 1 : (B11..B00)	Z DAC Output to drive piezo (12 bit)	0..10,000 mV
	X5	0	X5=0	Z DAC applied to Z piezo	Z DAC APPLIED
		1	X5=1	Z piezo fully retracted	FULLY RETRACTED
[XY CONTROL]	X_OFS	0..255	CS14=0; D07..D00;A / B=1; WR=0	X Offset DAC value (8bit)	0..10,000 mV
	Y_OFS	0..255	CS15=0; D07..D00;A / B=1; WR=0	Y Offset DAC value (8bit)	0..10,000 mV
	ZOOM	0..255	CS14=0; D07..D00;A / B=0; WR=0	X Zoom gain DAC value (8bit)	
			CS15=0; D07..D00;A / B=0; WR=0	Y Zoom gain DAC value (8bit)	
	XFBK_P	1..255	CS6=0; D07..D00;A / B=1; WR=0	X Feedback proportional DAC value	255..1
	XFBK_I	0..255	CS6=0; D07..D00;A / B=0; WR=0	X Feedback integral DAC value	
	YFBK_P	1..255	CS7=0; D07..D00;A / B=1; WR=0	Y Feedback proportional DAC value	255..1
	YFBK_I	0..255	CS7=0; D07..D00;A / B=0; WR=0	Y Feedback integral DAC value	
	XPI_ON	0	X13=0	X Feedback open loop	OFF
		1	X13=1	X Feedback closed loop	ON
	YPI_ON	0	X21=0	Y Feedback open loop	OFF
		1	X21=1	Y Feedback closed loop	ON
	EXTRA_ZOOM	1	none (Software flow control)	1x Extra Zoom on DAC - zoomed out	1x
		2	none (Software flow control)	2x Extra Zoom on DAC - zoomed in	2x

		4	none (Software flow control)	4x Extra Zoom on DAC - zoomed in	4x
	EXTRA_XOFS	0..4095	none (Software flow control)	Extra X Offset on DAC	0..10000mV
	EXTRA_YOFS	0..4095	none (Software flow control)	Extra Y Offset on DAC	0..10000mV
[FREQUENCY SYNTH]	FREQ	0..4,294,967,295	CS0,9=0;(D15..D00; WR=0/1)-MSW (D15..D00; WR=0/1)-LSW CS0,9=1 TC0..TC2=0; TC3=1;LOAD =1	Modulator frequency (32 bit)	0-20 MHz
	F_AMP	0..512	CS0,9=0; (D15..D00=0;WR=0/1)-MSW (D15..D10=0;D09..D00;WR=0/1)-LSW; CS0,9=1; TC0..TC1=1; TC2=0; TC3=1; LOAD =1;	Modulator amplitude (10 bit)	0..10,000 mV
	PHASE	0..4095	CS0,9=0; (D15..D00=0;WR=0/1)-MSW (D15..D12=0;D11..D00;WR=0/1)-LSW; CS0,9=1; TC0..TC1=1; TC2=0; TC3=1; LOAD =1;	Modulator phase shift (12 bit)	0.00..360.00
[AUX 1&2]	AUX1	0..4095	D11..D00; CS11=0|1 : (E11..E00)	AUX1 DAC Output (12 bit)	0..10,000 mV
	AUX2	0..4095	D11..D00; CS12=0|1 : (F11..F00)	AUX2 DAC Output (12 bit)	0..10,000 mV
[STEPPERS]	MOTOR	0..7	D[08..15]=0; CS4=0|1	Select Stepper by D[MOTOR + 08]	
	PULSES	0..65535	(D[0..7]=0/1; CS4=0|1) * PULSES	Output clocks on bit D[MOTOR]	
	STEP_DIR	0	D[08..15]=0; CS5=0|1	Fwd - Set bit D[MOTOR + 08] = 0	FORWARD
		1	D[08..15]=1; CS5=0|1	Rev - Set bit D[MOTOR + 08] = 1	REVERSE
	STEP	0	D[0..7]=0; CS5=0|1	Full step (D[MOTOR]=0)	FULL
		1	D[0..7]=1; CS5=0|1	Half step (D[MOTOR]=1)	HALF
	PACKET	1..65535;1-def	none (Software flow control)	Number of steps between network IO	1..65535
[LASER]	LASER	0	X3=0	Laser off	OFF
		1	X3=1	Laser on	ON
[DC MOTOR]	DCMTR_FWD	-128..127	CS1=0; D07..D00; A/B=1; WR=0	Z DC motor DAC value (8bit)	+/-5,000 mV
	DCMTR_REV	-128..127	CS1=0; D07..D00; A/B=1; WR=0	Z DC motor DAC value (8bit)	+/-5,000 mV
	DCMTR_TIME	100..65535	none (Software flow control)	Z DC motor "ON" time (ms)	100..65535 ms
[SCAN IMAGE]	POINTS	10..1500	none (Software flow control)	Resolution - points per line	16;..512;1024
	LINES	10..1500	none (Software flow control)	Resolution - lines per image	16;..512;1024
	SCAN_RATE	double floating point	none (C1&C2 ACL pacer dividers)	Scan rate - lines per second	0..(TBD)

	ROTATE	-360.00..+360.00	none (XY software calculation)	Scan area rotation angle	-360..360 deg.
	XYMODE	0	none (Software flow control)	Calculate XY on Master	on Master
		1	none (Software flow control)	Calculate XY on Slave	on Slave
	DIR	0	none (Software flow control)	ADC acquires data during fwd line scan	Forward
		1	none (Software flow control)	ADC acquires data during rev line scan	Reverse
		2	none (Software flow control)	ADC acquires data both fwd/rev scan	Fwd/Rev
	CH	1..4	none (Software flow control)	ACL acquires CH channels	
	DATA	0	none (Software flow control)	Transfer scan data after each line	by Line
		1	none (Software flow control)	Transfer scan data after whole scan	Whole
	SKEW	-10.00..+10.00	none (Software flow control)	Skew correction angle	-10..+10 deg
	OVERSCAN	0..127	none (Software flow control)	Number of overscan points	0..127
	PRESCAN	0..127	none (Software flow control)	Number of prescan lines	0..127
[TIP APPROACH]	ZMTR_TIP	0	none (Software flow control)	Select Z DC motor for tip retract/engage	Z DC motor
		1..8	none (Software flow control)	Select Stepper for tip retract/engage	Stepper 1..8
	CH_TIP	0..10	(see [INPUT SELECTS] section)	Select input channel to monitor	
	SRF_TIP	-32768..32767	none (Software flow control)	Select "close to surface" input value	+/-10000 mV
	DEV_TIP	328..32767	none (Software flow control)	Deviation from the "Surface" value	100..10000 mV
	DCREV_TIP	-128..127	CS1=0;D07..D00;A/B=1;WR=0	DC Motor reverse - tip retract (8 bit)	+/-5000 mV
	DCTIME_TIP	100..65535	none (Software flow control)	DC Motor reverse "ON" time	100..65535 ms
	DCFWD_TIP	-128..127	CS1=0;D07..D00;A/B=1;WR=0	DC Motor forward - tip engage (8 bit)	+/-5000 mV
	DIRUP_TIP	0	D[08..15]=0; CS5=0|1	Stepper direction forward - tip retract	FORWARD
		1	D[08..15]=1; CS5=0|1	Stepper direction reverse - tip retract	REVERSE
	STEPUP_TIP	0	D[0..7]=0; CS5=0|1	Stepper FULL step - tip retract	FULL
		1	D[0..7]=1; CS5=0|1	Stepper HALF step -tip retract	HALF
	PULSES_TIP	0..65535	(D[0..7]=0/1; CS4=0|1) * PULSES	Number of stepper pulses - tip retract	0..65535
	DIRDWN_TIP	0	D[08..15]=0; CS5=0|1	Stepper direction forward - tip engage	FORWARD
		1	D[08..15]=1; CS5=0|1	Stepper direction reverse - tip engage	REVERSE
	STEPDWN_TIP	0	D[0..7]=0; CS5=0|1	Stepper FULL step - tip engage	FULL
		1	D[0..7]=1; CS5=0|1	Stepper HALF step -tip engage	HALF
	CYCLES_TIP	1..65535; 500-deflt	none (Software flow control)	Number of acquisition cycles per step	1..65535
	PACKET_TIP	1..65535;1-default	none (Software flow control)	Number of steps between network IO	1..65535
	TIP_INC	0	none (Software flow control)	Generic Tip Approach routine	
		1	none (Software flow control)	Incremental Tip Approach routine	Incremental
	RUN_INC	1..65535; 20-deflt	none (Software flow control)	Number of half-steps per increment	steps

	RAMP_INC	1..4095; 20-default	none (Software flow control)	Z-DAC ramp increment	mV
	DELAY_INC	1..65535; 20-deflt	none (Software flow control)	Delay per Z-DAC ramp increment	ms
[FREQ SWEEP]	FREQ_S	0..4,294,967,295	CS0=0;(D15..D00; WR=0/1)-MSW (D15..D00; WR=0/1)-LSW CS0=1 TC0..TC2=0; TC3=1;LOAD =1	Start frequency (32 bit)	0-20 MHz
	FREQ_E	0..4,294,967,295	CS0=0;(D15..D00; WR=0/1)-MSW (D15..D00; WR=0/1)-LSW CS0=1 TC0..TC2=0; TC3=1;LOAD =1	End frequency (32 bit)	0-20 MHz
	F_AMP	0..512	CS0=0; (D15..D00=0;WR=0/1)-MSW (D15..D10=0;D09..D00;WR=0/1)-LSW; CS0=1; TC0..TC1=1; TC2=0; TC3=1; LOAD =1;	Modulator amplitude (10 bit)	0..10,000 mV
	F_PHASE	0..4095	CS0,9=0; (D15..D00=0;WR=0/1)-MSW (D15..D12=0;D11..D00;WR=0/1)-LSW; CS0,9=1; TC0..TC1=1; TC2=0; TC3=1; LOAD =1;	Modulator phase shift (12 bit)	0.00..360.00
	SWEEP_RATE	1..65,535;10 - def	none (Software flow control)	Sweep rate in ms per point	1..65,535 ms
[OSC TIME]	TIME_BASE	10..1000	none (Software flow control)	Acquisition time period	10..1000 ms
[OSC STORAGE]	TIME_BASE	2000..10000	none (Software flow control)	Acquisition time period	2000..10000ms
	DUTY_TIME	0..1900	none (Software flow control)	Calibration time	0..1900ms
[XYZ SCALE]	X_SCALE	double float	none (Master software only)	The fully zoomed out X scale size	in X_UNITs
	X_UNIT	0	none (Master software only)	microns	um
		1	none (Master software only)	nanometers	nm
		2	none (Master software only)	angstroms	A
		3	none (Master software only)	arbitrary units	
	Y_SCALE	double float	none (Master software only)	The fully zoomed out Y scale size	in Y_UNITs
	Y_UNIT	0	none (Master software only)	microns	um
		1	none (Master software only)	nanometers	nm
		2	none (Master software only)	angstroms	A
		3	none (Master software only)	arbitrary units	
	Z_SCALE	double float	none (Master software only)	The full Z scale size (Z-HGT)	in Z_UNITs
	Z_UNIT	0	none (Master software only)	microns	um
		1	none (Master software only)	nanometers	nm
		2	none (Master software only)	angstroms	A
		3	none (Master software only)	arbitrary units	
		4	none (Master software only)	millivolts	mV
	ZS_SCALE	double float	none (Master software only)	The full Z scale size (Z-SEN)	in ZS_UNITs
	ZS_UNIT	0..4 (as above)	none (Master software only)	microns, nanometers, A, arbitrary, mV	um,nm, A, ,V
	AUX1_SCALE	double float	none (Master software only)	The full Z scale size (AUX-IN1)	in AUX1_UNITs
	AUX1_UNIT	0..4 (as above)	none (Master software only)	microns, nanometers, A, arbitrary, mV	um,nm, A, ,V

	AUX2_SCALE	double float	none (Master software only)	The full Z scale size (AUX-IN2)	in AUX2_UNITs
	AUX2_UNIT	0..4 (as above)	none (Master software only)	microns, nanometers, A, arbitrary, mV	um,nm, A, ,V
	ADC_SCALE	double float	none (Master software only)	The full Z scale size (Other ADC)	in ADC_UNITs
	ADC_UNIT	0..4 (as above)	none (Master software only)	microns, nanometers, A, arbitrary, mV	um,nm, A, ,V
[TIP XY]	X_POS	0..4095	none (Software flow control)	X-DAC voltage	0..10000 mV
	Y_POS	0..4095	none (Software flow control)	Y-DAC voltage	0..10000 mV
	PID_ON	0	X4=0	Turn PID OFF before moving	OFF
		1	X4=1	Turn PID ON before moving	ON
[IDLE PARK]	ENABLE	0	none (Software flow control)	Disable Park Idle Piezos feature	Disable
		1	none (Software flow control)	Enable Park Idle Piezos feature	Enable
	TIME	0..65535	none (Software flow control)	Idle Timeout in minutes	min.
[FORCE DIST]	ZDAC_S	0..4095	none (Software flow control)	Start value for Z-DAC output	0..10000 mV
	ZDAC_E	0..4095	none (Software flow control)	End value for Z-DAC output	0..10000 mV
	CH	0..11	(see [INPUT SELECTS]:CH1)	ADC signal channel for force / dist.	(see CH1)
	FORCE_RATE	0..65535	none (Software flow control)	Rate in ms per point	ms / point
	PIX	16..1024	none (Software flow control)	Resolution in pixels per each curve	16;..512;1024
	DEF_LIMIT	-32767..32767	none (Software flow control)	Deflection Limit value	- / +10000 mV
	NUMBER	1..100	none (Software flow control)	The number of curves to average	1..100
	CONTINUE	0..1	none (Software flow control)	Continuous acquisition mode	Ena / Dis
	F_SCALE	double float	none (Master software only)	Force Calibration - full scale	in F_UNITs
	F_UNIT	characters	none (Master software only)	Force Calibration units - name	Characters
	SPRING_K	double float	none (Master software only)	Cantilever Spring Constant	nN / nm
[NON LINEAR]	SCL_CORR	0	none (Master software only)	Disable Scale Nonlinearity Correction	Disable
		1	none (Master software only)	Enable Scale Nonlinearity Correction	Enable
	SX1	double float	none (Master software only)	Xreal = Sx1*Xsen +	none
	SX2	double float	none (Master software only)	+ Sx2*Xsen*Xsen +	V^(-1)
	SX3	double float	none (Master software only)	+ Sx3*Xsen*Xsen*Xsen	V^(-2)
	SY1	double float	none (Master software only)	Yreal = Sy1*Ysen +	none
	SY2	double float	none (Master software only)	+ Sy2*Ysen*Ysen +	V^(-1)
	SY3	double float	none (Master software only)	+ Sy3*Ysen*Ysen*Ysen	V^(-2)
	IMG_CORR	0	none (Software flow control)	No image nonlinearity correction	No
		1	none (Software flow control)	On-line image correction (acquire)	On-line
		2	none (Master software only)	Off-line image correction (resample)	Off-line
	A1	double float	none (Software flow control)	xsen = A1*xreal +	none

	A2	double float	none (Software flow control)	+ A2*xreal*xreal +	pixels^(-1)
	A3	double float	none (Software flow control)	+ A3*xreal*xreal*xreal	pixels^(-2)
	B1	double float	none (Software flow control)	ysen = B1*yreal +	none
	B2	double float	none (Software flow control)	+ B2*yreal*yreal +	pixels^(-1)
	B3	double float	none (Software flow control)	+ B3*yreal*yreal*yreal	pixels^(-2)
	X_OFS0	0..255	none (Software flow control)	X offset of calibration image	0..10,000 mV
	Y_OFS0	0..255	none (Software flow control)	Y offset of calibration image	0..10,000 mV
	ZOOM0	0..255	none (Software flow control)	Zoom of calibration image	
	PIX0	128..1024	none (Software flow control)	Resolution of calibration image	pixels
[HYST CORR]	ZOOM0	0..255	none (Master software only)	The zoom of reference scans	
	PIX0	128..1024	none (Master software only)	The resolution of reference scans	pixels
	OFS0	0..255	none (Master software only)	The offset of lower-right reference scan	mV
	OPTB0	double float	none (Master software only)	Hyst. Correction' 2nd order coeff. (1st reference)	
	OFS1	0..255	none (Master software only)	The offset of upper-left reference scan	mV
	OPTB1	double float	none (Master software only)	Hyst. Correction' 2nd order coeff. (2nd reference)	
	PX0	double float	none (Master software only)	X Scale Factor primary correction (vs. Zoom)	
	PX1	double float	none (Master software only)	Lx0 = px0 + px1*Zoom(LSB) + px2*Zoom(LSB)^2	
	PX2	double float	none (Master software only)		
	PY0	double float	none (Master software only)	Y Scale Factor primary correction (vs. Zoom)	
	PY1	double float	none (Master software only)	Ly0 = py0 + py1*Zoom(LSB) + py2*Zoom(LSB)^2	
	PY2	double float	none (Master software only)		
	CX0	double float	none (Master software only)	X Scale Factor secondary correction (vs. X offset)	
	CX1	double float	none (Master software only)	Lx = Lx0 + (cx0 + cx1*Xoffs(mV))*Xoffs(mV)	
	CY0	double float	none (Master software only)	Y Scale Factor secondary correction (vs. Y offset)	
	CY1	double float	none (Master software only)	Ly = Ly0 + (cy0 + cy1*Yoffs(mV))*Yoffs(mV)	
	A1	double float	none (Master software only)	Hysteresis correction via FFT - last known coeff.	
	A2	double float	none (Master software only)	Xreal = a1*Xpos + a2*Xpos^2 (pixels)	
	B1	double float	none (Master software only)	Yreal = b1*Ypos + b2*Ypos^2 (pixels)	
	B2	double float	none (Master software only)		
[AUTO LINEARIZER]	XOFS_ADJUST	-10,000..+10,000	none (Master software only)	X offset adjustment (default - 300 mV)	mV
	YOFS_ADJUST	-10,000..+10,001	none (Master software only)	X offset adjustment (default - 300 mV)	mV
	XPIX_ZOOM	float	none (Master software only)	X pixel-zoom factor (default - 0.9, 90%)	%
	YPIX_ZOOM	float	none (Master software only)	Y pixel-zoom factor (default - 0.9, 90%)	%

Appendix H: Software Specification Summary for SPM-Cockpit

Utility Program to Test Functional Groupings of Controller

MDI Window (Multiple Document Interface)

- Menu
- Toolbar
- Status Bar

Child Windows

OSCILLOSCOPE, TIME-MODE (UP TO 4 MODELESS WINDOWS)

- Select Channel
- Select Range
- Select Offset
- Select Time-base, Apply Button

OSCILLOSCOPE, FREQUENCY SWEEP MODE (1 MODELESS WINDOW)

- Select Channel to Monitor
- Select Frequency Range
- Select Modulation Amplitude
- Start Sweep/ Stop Sweep

OSCILLOSCOPE, SCAN LINE MODE (UP TO 4 WINDOWS MODELESS WINDOWS)

- Select Channel
- Select Range
- Select Offset
- Select Number of Pixels

DISPLAY SCANNED IMAGE (UP TO 4 MODELESS WINDOWS)

- Select Channel
- Select Forward/reverse

SCAN CONTROL PANEL (1 MODELESS WINDOW)

- Start Scan, Stop Scan

- Scan Progress Indicator and Elapsed Time

- Option: Show Oscilloscope, Line Mode and Display Scanned Image @ Same Time

RED DOT ALIGNMENT (1 MODELESS WINDOW)

TIP APPROACH (1 MODAL WINDOW)

Tip Engage

- Select Z Motor: Stepper or Dc Motor

- Select Channel to Monitor (See Selection Below)

- Select "Close to Surface" Value

- Select Z(ADC) Final Nominal Value

Tip Retract

- Select Z Motor: Stepper or Dc Motor

- Select Dc Motor Reverse Value

- Select Dc Motor Reverse Time

Tabbed-windows (Modeless)

INPUT SELECTS TO ADC
- SELECT UP TO 4 CHANNELS

- 11 Channels to Select (8 Bit Port Addr & 4 Bits)

- See Selection above

Z SENSOR SIGNAL SELECTS

- Gain (8 Bits)

- Offset (8 Bits)

- Filter (3 Bits for 10 Hz, 100 Hz, 1000 Hz, Full Range)

ERROR SIGNAL Z(POS) SELECTS

- Filter (3 Bits for 10 Hz, 100 Hz, 1000 Hz, Full Range)

Z(ADC) SELECTS

- Gain (1 Bit for 1x, 3x)

LATERAL FORCE SELECTS

- Gain (8 Bits)

- Offset (8 Bits)

- Filter (3 Bits for 10 Hz, 100 Hz, 1000 Hz, Full Range)

Z FEEDBACK

- PID Channel Select (2 Bit)
  - T-b Photodet or External Input or Z(sen)
- Select Demod Mode (2 Bit)
  - Demod or Bypass Demod
- Select Input Polarity (1 Bit)
- Select Setpoint Polarity (1 Bit)
- Select Setpoint Value (8 Bits)
- Select Error Signal Gain (8 Bits)
- Select PID Values (3 X 8 Bits)

Z PID ON/OFF STATE

- PID On/off (1 Bit)

DEMOD SELECTS

- Demod Gain (3 Bits)
- Demod Filter (3 Bits for 10 Hz, 100 Hz, 1000 Hz, Full Range)

Z PIEZO

- Output to Drive Piezo
- Z Dac out (12 Bits)

X-Y CONTROL

- X,y Offset (8 Bits Each Axis)
- Zoom (8 Bits)
- P-I Control of X and Y Feedback Loops (2 X 8 Bits Each Axis)

FREQUENCY SYNTHESIZER

- Frequency Select (32 Bits)
- Amplitude Select (10 Bits)
- Frequency Sweep W/ Oscilloscope Window

AUX 1 & 2 OUTPUTS

- Select Value (12 Bits, 0 - 10 V)

LASER/MOTORS

- Motor Select (6 Steppers @ 0.5 Amp / Phase)
- Test Routine
    - Direction, Full/half Step, Enable, Clock
- Laser On/off (1 Bit)
- Dc Motor Forward and Reverse Values (8 Bits), Test Routine
- Dc Motor Apply Time

SCAN IMAGE SETUP

- Resolution (Number of Points & Lines)
- Select Forward And/or Reverse Scan (Selected in Software)
- Scan Rate Select (Lines per Second)
- Scan Rotation Select (-360..+360 Deg)
- Two Software Options (Tbd)
    - Calculate X,Y Coord on Master
    - Calculate X,y Coord on Slave
- Couple of Issues, Re: Overscan Slow, Reverse, Accel.
- Select Number of Channels to Acquire (From 1 to 4)

Appendix I: TopoMetrix File Extension Assignments

Use of Topometrix Image File Types for PScan2™

PScan2™ Channel	PScan2™ Signal	File extension *		Topometrix File Type **	Comments
		Forward	Reverse		
0	Z(POS)	.zfr	.zrr	Topography or Topo	
1	Z(HGT)	.tfr	.trr	Topography or Topo	True topography
2	Z(LR)	.lfr	.lrr	Lateral Force or LFM	
3	Z(SEN)	.ffr	.frr	Feedback sensor or Sensor	
4	AUX(IN1)	.1fr	.1rr	External Input 1 or ADC1	
5	X(SEN)	.xfr	.xrr		Not recognized by Topometrix Software ***
6	Z(DEM)	.sfr	.srr	Modulation or Spectro	
7	Y(SEN)	.yfr	.yrr		Not recognized by Topometrix Software ***
8	Z(ERR)	.mfr	.mrr	Fast Track	
9	Z(SUM)	.ufr	.urr		Not recognized by Topometrix Software ***
10	AUX(IN2)	.2fr	.2rr	External Input 2 or ADC2	
11	ADC8B	.8fr	.8rr	ADC8B auxiliary input	Not recognized by Topometrix Software ***

*) The first letter of a file extension represents the signal source, the second - line acquisition direction (forward -"f", reverse - "r"), the third letter represents the image origin (raw - "r", processed - "p")

**) As appears in Topometrix Software

***) In order to be processed by Topometrix Software the file must be manually renamed to a file name with one of the recognized file extensions. Select a Topometrix extension that does not currently hold data already acquired by PScan2™ controller.

Appendix J: Flow Chart for "SPM-Cockpit" Program (Located in Separate Volume)

USER

Typical Mode

This mode is intended for the occasional user. It provides access to a limited number of functions on the PScan2™ Controller. These functions are arranged as a sequential menu and offer a step-wise method for initializing the scanner, setting scanning mode, beam alignment, tip engagement and retract, selected scanning options, image display and direct access to an image analysis program (optional program). This mode will over-ride certain parameters in the configuration file, using instead "default" parameters that minimize the possibility of an improper cantilever/tip engagement.

Expert Mode

This mode provides access to all the functions currently available on the PScan2™ Controller.

FILE

Open Configuration File

The configuration file contains the operating parameters that have been previously set and saved. Any directory can be accessed; the default directory can be set under "Preferences". The SPMCockpit™ program opens with the setup and scanning parameters from the previous session.

Save Configuration File as ...

This function will save the setup and scanning parameters in the current session at any time.

Edit Configuration File

Under some conditions it may be useful to manually change a particular parameter in the Configuration File. A text editor is opened to allow changes. The file must be saved when exiting the editor in order for the changes to take place.

Save Images

An image is acquired for each of the selected channels (see "Scan Image Setup"), 1 through 4. They may be saved in one or any of three formats: TopoMetrix (ThermoMicroscopes), Nanoscope 3 (Digital Instruments, Veeco) and Digital Surf, which can be accessed by the corresponding image analysis software. A list of extensions for each type of image acquired is provided in the appendix of the User's Manual. Please note that in the "Typical Mode" only one image, the "clicked-on"

highlighted image can be saved in the image bank under “scanner controls”. However, all the images can still be saved under the “file” menu selection, as in the “Expert Mode”.

Open Images
Images of any of the three above mentioned formats may be opened for viewing. The image can then be saved in another format or exported (discussed below).

Save Raw Scan Data
The primary scan data may be saved in a “Pacific Nanotechnology” format.

Open Raw Scan Data
This function opens a “Raw Scan Data” file for viewing. The image may then be saved in another format or saved in a standard format for exporting into other word processing and spreadsheet program files.

Export Displayed Images
Images may be saved for export as Bitmap (*.bmp), GIF (*.gif), JPEG (*.jpg), or TIF (*.tif) files.

Preferences:

Configuration
The user may select a default directory for storing and retrieving Configuration (*.cfg) files.

Raw Data
The user may select a default directory for storing and retrieving “Raw Data” files.

Export Image
The user may select a default directory for storing and retrieving “Image” files (.bmp, .gif, .jpg, .tif formats) for exporting into other programs.

Image Files
The user may select a default directory for storing and retrieving “Image” files in TopoMetrix, Nanoscope, and Digital Surf formats.

Auto-linearizer
The user may select the size of the lower and upper “buffer” regions for use by the “auto-linearizer” routine. The size of the buffers for the X and Y axis may be set separately. The lower buffer is set in “millivolts” which is typically 200 to 700 mV. The smaller the value, the smaller is the buffer region. If the setting is too small, that the lower voltage side of the scanned image (depending on rotation angle) will appear distorted along that axis. The upper buffer is set as a percentage, and is typically 90 to 96%. If the value is set too high, then the higher voltage side of the scanned image will be distorted for that axis. Smaller mV and higher percent value will increase the scan range. However, thermal

effects and piezo performance may cause small changes in the operating range of the actuators, requiring more frequent use of the auto-linearizer function.

Miscellaneous

The user may select several among display and activity options for operating convenience.

Typical Mode

The user may select certain properties as to how the Typical Mode is displayed and as to which image analysis program is activated.

Calculate Non-linearity

This function is reserved for qualified technical persons.

Exit

This function closes the image acquisition program.

DEVICE

Directory Setup

The Input/output (I/O) directory for a particular PScan2™ Controller may be selected from this menu item. Be sure to "double-click" on the directory of choice in order to bring the directory name into the lower window before clicking "OK".

Create Device Directory

When first installing a new PScan2™ Controller to the Master Computer, a new directory must be created. Usually, the name of the directory is the same as serial number of the Controller. For example, the directory can be named "PscanXXX", where "XXX" is the last three digits of the serial number of the Controller.

Create Configuration Diskette

Before the PScan2™ Controller can recognize the existence of the Master Computer on the Ethernet, a Configuration Diskette must be generated by the Master Computer and installed on the Slave. Using an empty diskette, follow the instructions displayed in the window. Remember to disconnect the Ethernet cable to the Controller before rebooting the Controller in order for the Controller to accept and read the diskette generated from the Master Computer. This operation takes only a few minutes. A series of four medium-length beeps indicates that the diskette was read and information transferred. Also, remember to reconnect the Ethernet cable when the configuration file installation is complete and the Controller is rebooted. A "hi-lo-hi" series of three beeps indicates that the Controller has been successfully connected to the Ethernet. This may be confirmed by using the "Ping" function.

Ping

This function provides a quick means for confirming that the PScan2™ Controller is connected to the Master Computer and is operating properly. If network and controller are working there is an almost immediate response indicating that the system is operational. A time-out error message is displayed after a few seconds if the system is not working.

Multiple Unit Setup

Two or more controller/scanners can be operated from one master workstation. This function allows the user to set-up one program to operate two controllers independently by setting a "multiple device capability". During operation simply selecting the "device menu" can operate each controller/scanner.

Park Idle Piezos

The performance characteristics of the piezo actuators may deteriorate if substantial voltage is applied to the actuators over an extended period. This function provides a "time-out" capability when the controller is in Idle Mode; that is, when there is no active data acquisition (such as oscilloscope or image acquisition modes). The piezo voltages are brought to zero volts after a specified time period. If the tip is in feedback, a "tip retract" operation will be performed. Any movement of the mouse will reactivate the actuator voltage setting, but the tip engagement routine will not be performed. The minimum time-out range is 1 minute; the maximum is 1000 minutes.

Diagnostics

This function is reserved for qualified technical persons.

SETTINGS

Input Selects to ADC:

Channel Selects

Up to four channels of twelve possible signals may be monitored using the various oscilloscope functions and scan image functions (see the User's Manual for a list and description of these signals). Certain channels, ones in which simultaneous monitoring is not typically needed, cannot be acquired at the same time. If a conflict during selection occurs, an error message will be displayed. One or more channels are only available for oscilloscope line display and image acquisition if the appropriate numbers of channels are selected in the "Scan Image Setup" section. All four channels are available in the oscilloscope time mode.

Lateral Force

Normally, a gain of "one" and offset of "255" is sufficient for most lateral force imaging situations. If a higher gain is needed, the "Red Dot" should be set to the left of the vertical mid-line, near the left border of the green zone. The gain and offset can then be adjusted while scanning for optimal image acquisition. An optimal filter setting (full range, 1000 Hz, 100 Hz and 10 Hz) can also be established.

Z Sensor, Z(Sen)

These parameters are only active for 3-axis sensor scanners. The Z sensor gain and offset are factory calibrated to approximately overlap the Z(Hgt) span in microns (7-10 microns). For correct ranging, the gain is typically set at "7", and the offset is adjusted so that the center of the Z(Hgt) range (zero volts) and the center of the Z(Sen) range are approximately the same. The decade filter allows for more precise height measurements, although at lower scan rate.

Error Signal Z(Pos) Filter

The Z(Pos) signal is the inverted Z(Err) with the addition of decade filters (full range, 1000 Hz, 100 Hz, and 10 Hz). This signal is useful for reducing noise and noise spikes in either contact mode or oscillating mode.

Z(Hgt) Gain

The Z (Hgt) bit resolution can be doubled, particularly for enhancing the image quality of features of small height. The Z range is also reduced a factor of two, with the "extended" half of the range being active (0 to 10 V of the -10 to +10 V).

Z Piezo

With the PID setting "off" and the "Fast Retract" set to "Z DAC applied", the Z piezo voltage may be set directly with this function. The voltage range of the piezo is the inverse of the set voltage (10,000 to 0 mV for 0 to 130 (140V in newer units)) V output to the Z piezo. The "fully retracted" setting of the "fast retract" function is intended for a specialized application; the normal setting is "Z DAC applied".

PID On/Off

The PID loop may be turned on or off manually at any time. In any event, during tip approach the PID is turned on before approach motor activity. During tip retract, the PID is turned off after the motor pulls back.

Scan Image Setup

Resolution

Over all scan ranges the images may be acquired at resolutions from 16 x 16 to 1024 x 1024 pixels.

Overscan

Under some conditions, it may be useful to reduce artifacts associated with reversing tip scan direction. When scanning in the forward direction, up to 127 pixels may be "removed" at the beginning of the line. Depending on the resolution, the range will be reduced proportionally. The resulting image is still at the specified scanning resolution.

Prescan

As above, under some conditions it is useful to remove a few lines at the beginning of a scan. Up to 127 lines may be eliminated from the acquired image. Again, the resulting image is still at the specified scanning resolution.

Scan Rate

The scan rate may be varied from a few thousandth of a line per second up to about 15 lines per second for four channel acquisition. Above 0.2 lps, user interrupts are only allowed at the end of a scanned line; interrupts are allowed after each point below 0.2 lps.

Channels

One to four channels may be acquired simultaneously and stored.

Skew

The extent of skew between the X & Y axis is corrected during image acquisition. It is typically less than one degree.

Rotation

The angle for the fast-scan axis can be set manually or by clicking on one of the primary axis setting. The slight distortion effect of the skew correction setting in the first few scanned lines along the X-axis may be removed by setting the rotation angle the same as the skew angle; thereby, the "zero" angle for scanning becomes the skew angle.

Scan Data Transfer

Image data can be transferred form the Controller to the Master Computer on a line-by-line basis or after the whole image is acquired.

Scan Image Direction

The image may be acquired in the Forward, Reverse or Both directions of the fast axis.

X-Y Control

The **X-Offset**, **Y-offset** and **Zoom** set the effective scan range and start-of-scan position, which is also dependent on the rotation angle. The initial offsets and maximum zoom are set by the configuration file or by running the Auto-linearizer routine (see below). These values can be adjust for smaller scan range and location by using the two zoom features: "box-click" mouse on the image for zoom-in, or "box-click" mouse adjacent to the image for zoom-out; or "double-click" adjacent to the image to open a separate window for zoom-in or zoom-out. Alternatively, the user may adjust these boxed values to select a particular value.

In addition to the zoom features above, an Extra Zoom with Offsets can be selected for higher resolution or positioning capability. By "double-clicking" within an image, a new "extra Zoom" window is opened. The user may select a zoom value of 2X or 4X and locate scanning region.

The X & Y Feedback parameters are also set within this section. The respective feedback loops may be manually turned on or off for test purposes.

Z-feedback:

PID Channel
One of three inputs may be selected for the Z-Feedback Loop: (1) T-B Photodiode from the AFM scanner, (2) External and (3) Z-Sensor (if option is installed).

Input Polarity
The PScan2™ Controller is capable of engaging in feedback for both positively going and negatively going error signals. Typically SPM users are more familiar with the former; i.e. with the error signal response from a lower (more negative) signal to a more positive signal. For contact mode, the Input Polarity is set for "Positive". For the oscillating mode, it is set for "Negative".

Set-point Polarity
For contact mode, the Set-point is typically set to "zero" and the Set-point Polarity setting is not important. If a positive Set-point value is required, the setting is "Positive, and visa versa if a negative Set-point value is needed. For oscillating modes, the Error Signal is always negative (see above); the Set-point Polarity is "negative".

Set-point Value
The Set-point range is from 0 to 10 volts; the sign depends on the Set-point Polarity. For contact mode the Set-point, typically set for "zero" volts initially, may be increased or decreased to change the force on the cantilever. For the oscillating modes involving a resonance of the cantilever (a "negative" error signal, see above), the Set-point is usually "negative". As an example, the setting ranges from 0.5x to 0.7x of the Error Signal out-of-feedback for "hard" intermittent contact mode to 0.8x to 0.9x for "light" intermittent contact mode. The actual preferred setting depends on the cantilever/tip characteristics as well as the particular experiment at hand.

Demodulation
"Bypass" is selected for contact mode; "Demod" activates the demodulator circuits and is selected for any oscillating mode.

Error Signal Gain
The Error Signal Gain compensates for variations in cantilever reflection intensity. For contact mode, if Z(Sum) is at or near "max" (about 40% of full scale), the Error Signal Gain is typically "1", increasing to a value of "10 - 15" for Z(Sum) at its minimal value. The Error Signal Gain ranges from 1 to 255.

PID Values

P (Proportional), I (integral) and D (Derivative) gains may be set from 1 to 255. The higher the value, the greater is the influence of the parameter. A typical setting for contact or intermittent contact modes is P=8, I=20, D=1.

Frequency Synthesizer

Frequency

The frequency of the driving oscillator for the oscillating modes may be set from a few Hertz to several MHz. The typical range for this system is from 50 kHz to 500 kHz.

Amplitude

The peak-to-peak amplitude may be set from a few millivolts to 10 volts in approximately 20 mV increments.

Phase

For Phase Detection, varying the phase of the detected signal relative to the reference signal may enhance the contrast of the acquired image. The range is 0 to 360 degrees.

AUX 1 & 2 Outputs

The Auxiliary outputs, AUX1 and AUX2, provide the user with 0 - 10 V DC, 10 mA outputs with 12 bit resolution.

Demod Selects

Demod Gain

Four settings are available: 1x, 2x, 3x, and 4x. The higher gains are used to assure a satisfactory signal level for the demodulator section.

Demod Filter

High frequency effects can be reduced after demodulation by the use of decade filters: 10 Hz, 100 Hz, 1000 Hz, and full range.

Demode Type

This switch selects the type of detection in the demodulator: Amplitude Detection (typical for oscillating mode) and Phase Detection (used to control the PID loop by sensing small changes in the phase of the oscillating cantilever/tip).

Laser/Motors

Laser

The Laser is ON under typical program initialization. It can be turned off at any time.

Stepper Motors

Motor Select

Any of six micro-stepper motors (rated at 12 V, 0.5 A per phase) can be selected. Just to the right of the select window is a window that indicates the motor location on a relative scale. The primary motor for AFM is motor #1.

Control

An individual stepper may be moved in the Forward or Reverse direction and stopped at any time. Any number of Pulses may be sent to the motor; the size of the packet number determines the pulses that the controller sends to the stepper motor as a "packet". This packet method for pulse transfer allows the controller to check for a user interrupt between packets, such as "STOP". The Step size may be Full or Half Step.

DC Motor

Some scanners incorporate a +/- 5 V DC motor for controlling the tip/cantilever approach and withdrawal. The Forward and Reverse voltages are controlled separately (in mV) for a given Duration (in msec). The Stop button sets the voltage to zero.

X Y Z Scales

X & Y Full Scale

The X & Y axis scales may be set independently and can be expressed in Angstroms (A), nanometers (nm) or microns (um). If the two axes are not in feedback, the numbers entered represent the actual "full scale" of the scanned image. For systems with X & Y feedback and linearization, the situation is more complicated, resulting in the need to enter larger numbers than the actual scan range.

Z Full Scale

The Z(Hgt) represents the voltage applied to the actuator in the Z PID loop in order to maintain feedback. Z(Sen) is the output of the optional independent sensor. The scales for both signals may be presented in the three units mentioned above. All other ADC voltages are given the same full-scale factor and units.

Non-linearity X Y

To correct for non-linearity of the X & Y actuators or linearization sensors, several options are available for up to third-order correction.

Scale Correction

With the Image Correction turned off, and the Correct box checked, the image scale is corrected to positional non-linearity for any zoomed region within the full scan area.

Image Correction

Off-line A correction is applied to an uncorrected scanned image. The data is then re-sampled when stored in the Digital Surf format. The **Correct** box of the Scale Correction may be either on (conveniently displaying the proper range when scanning) or off.

On-line A correction is applied during scanning, eliminating the need for off-line correction. The actual correction parameters for all options are factory-set.

TOOLS

Red Dot Display

This display provides a convenient means for aligning the laser beam onto the detector. For contact-mode AFM, the red dot is located below the horizontal median line within the green region. The Set-point is typically set at zero volts, so that the red dot crosses the median line in an upward direction as the cantilever/tip contacts the surface. The more negative (lower) the red dot, the higher the contact force. If Lateral Force images are to be acquired, which may require increased gain settings for Lateral Force (see Settings), then the red dot should be set slightly to the left of the vertical median line and below the horizontal median line.

The bar meter to the right of the red dot region shows the total light intensity on the detector, Z(Sum). For more precise beam positioning at low light levels, a 1x to 4x Scale switch is provided. This scale setting does not affect the Z-PID loop.

For convenience, the Laser may be turned on and off from this window.

Oscilloscope, time mode

The time dependence of up to 4 signals may be presented graphical form. The time-base ranges from 10 to 1000 ms. The update interval depends on the time-base setting and the performance of the master computer, but is typically several times a second.

As with all oscilloscope modes, the voltage range is +/- 10 volts. The scaling may be set at Full scale, one-time Auto-scale or continuous auto-scale (with box checked). With the Auto box unchecked, the **Half Range** and **Offset** settings can be independently controlled.

Oscilloscope, line mode

This mode is similar to the time mode, except that the abscissa becomes the voltage ramp of a line scan. The repetition rate is set under the **Scan Image Setup** (scan rate) as is the resolution (pixels).

If the Z(Hgt) or Z(Sen) signals are selected, an **Auto-leveling** box is available for observing the line scan corrected for background slope.

Oscilloscope, frequency sweep mode

This mode is for determining the characteristics of a demodulated signal as a function of driving frequency and amplitude. It provides a convenient means for frequency scanning (nominally 50 kHz - 500kHz), determining the resonance frequency, setting the driving frequency and amplitude, and measuring the signal amplitude as a function of applied Z-piezo voltage. The scaling boxes are similar to the modes above.

To the lower left of the graph is the selected Signal. The Sweep Rate is typically set to 5 - 10 ms. The Start and End frequencies are typically set around the anticipated resonance frequency, or they may be set to scan the full range. The Start and Stop Sweep initiate and terminate the frequency sweep. The driving amplitude and phase (for phase-detection mode) may be set at anytime. Two successive sweeps are displayed: the current sweep is green; the previous sweep is red.

For convenience, the left mouse key may be pressed while the cursor is within the graph screen in order to sweep a particular range. The range is fixed and the graph screen reset to the new sweep ranges by a right mouse click. Positioning the cursor on the desired frequency and double-clicking the left mouse button sets the indicated frequency, amplitude and phase into the Settings windows when the user is ready for tip/cantilever approach.

The "Z>" button is used to ascertain whether the cantilever and mount are in satisfactory mechanical contact as a function of the applied Z-piezo voltage. Once the desired frequency is selected, pressing the "Z>" button will ramp the output voltage to the Z actuator from high-to-low-to-high voltage and return to the initial state. The resulting plot of the amplitude at the selected resonance frequency should be relatively flat (less than 10% variation). This assures that the amplitude is substantially constant over the Z actuator range.

When operating in the Typical Mode, two additional functions are presented: Full Range sets the sweep range from 50kHz to 400kHz. Autoset sets the frequency at 90% of the maximum value to the left of the peak and sets the setpoint so that error signal (Z(Err)) is zero at two-thirds of maximum amplitude.

Detector Sensitivity: The amplitude of the cantilever oscillation, under typical conditions for cantilevers in the 250 - 350 kHz resonance frequency range, is

approximately 0.12 nm/mV drive amplitude for Nanosensor cantilevers and approximately 0.31 nm/mV drive amplitude for Ktek cantilevers.

Dual-trace Storage Scope Mode

This mode is similar to the time mode, except that the full-scale time ranges from 2000 to 10000 ms. Although only two signals may be displayed at any time, the two signals are synchronized to within 15 to 45 microseconds, depending on which channels are selected. This mode is useful for observing long term drift effects.

Automatic Linearizer

This function automatically maps the X and Y actuator movements onto a selected region within the sensors' full scale range. From the Preferences window, the lower limit (selected as a millivolt offset) and the upper limit (selected as a percentage of the maximum available range) for each axis provide a buffer zone, below the lower limit and above the upper limit, in order to assure that the signal in the feedback loop will not exceed the actuator's mechanical range. There is a succession of eight windows presented in order to provide a visual sense as to how well each axis is operating.

Tip Approach / Retract

The primary control of engaging and retracting cantilever/tip relative to the scanning surface are three icon buttons. **Stop** is in the center of the window, **Retract** is just above, and **Approach** is just below the Stop icon. The motor for Z motion is selected in the box just to the right of the Stop icon. If a stepper motor is selected, then the relative position of the tip/cantilever is shown in the box just below the motor selection box.

Approach

For systems using DC motor, the Voltage box sets the rate of approach. The voltage may be positive or negative. For systems using Steppers, the Step Size and Direction may be selected. The approach rate is fixed, but may be changed if needed (See User's Manual, DCEx™, initialization file structure).

Select Channel

Any signal sensing the Z cantilever/tip interacting with the surface, i.e., the feedback error signal, may be selected. Typically, it is Z(Err) for contact mode and Z(Dem) for oscillating modes.

"Surface" Value & Deviation

In order to provide for both positive-going and negative-going feedback error signals as the tip/cantilever approaches the surface, the motor can be stopped within any range of positive or negative signal voltage. The Surface Value is the center of the required voltage range, and Deviation is the voltage above and below the center voltage. For example, for most AFM systems, the feedback error signal is negative-going-positive, with zero volts as the cross-over, i.e., the voltage above which will stop the approach motor. A Surface Value and

Deviation setting of 5000 mV and 5000 mV, respectively, will stop the motor if the feedback error signal is between zero and +10 V.

Incremental Approach

This function activates a Z-voltage ramp during tip approach that provides a more gentle interaction of the tip with the surface during tip engagement. Prior to each downward motion of the stepper motor (a few microns), the Z-piezo is ramped at a set rate with the PID feedback loop off. The process is stopped when the Z(Err) becomes positive, and PID loop is activated. If the recommended parameters (displayed when the Advanced window is opened), are used, the tip will be positioned approximately mid-scale for Z(Hgt). The parameter settings, listed from top to bottom, are: Fast Approach-500,50,20,20,1; Medium - 500,200,20,20,1; and Slow - 500,500,20,20,1. The approximate approach rates for each speed (overall rate followed by actual ramp rate in microns per second): Fast - 2.0/ 0.9; Medium - 0.9/ 1.5; Slow - 0.5/ 0.6

Monitored Value

As the tip/cantilever approaches the surface, the feedback error signal is periodically updated in this box.

Retract

For DC motors, the extent to which the cantilever/tip is retracted from the surface is set by a combination of the applied Voltage and the Duration in ms of the voltage pulse.

For steppers, the Step Size, Direction, and Number of steps are set. The rate is preset at the maximum rate for reliable operation. The Distance movement of the cantilever/tip is indicated in the box adjacent to the step number.

Scan Control Panel

Scanning and image acquisition is controlled from this window with the **Start** and **Stop** buttons. If the **Repeat Scan** box is checked, the scan routine will restart a few seconds after completion. The **Elapsed Time** and **Lines Remaining** for the scan are updated during scanning.

Display Scanned Image

By successively clicking on the "grid" icon, up to four different signals can be imaged during scanning. (See the **Settings** section for selecting available signals). Any of the displayed images may be expanded to full screen.

The color bar on the left of the image represents the range of signals for the data acquired. Checking the **Histogram Correction** box allows the user to define the Z scale of interest, with the upper and lower limits set as a percent of full scale. For some

viewing situations, the quality of the image may be enhanced by checking the **Shading box** and selecting the apparent direction of the light source (**N, S, E, & W**).

The image can be leveled on a line-by-line basis by checking the **Auto-leveling** box. The color bar and scaling are corrected automatically.

Zoom

An area to be zoomed may be defined by pressing the left mouse button on the upper left region to be outlined and drawing the cursor across the scanned image. The zoomed area is locked-in by releasing the left mouse button and clicking on the right button. To zoom out to a previously zoomed region, a small box is formed just outside the scanned image, but within the window. This procedure may be performed successively, expanding the zoomed area until the maximum scanned area is accessed.

An alternative means for zooming is accomplished by double clicking the left mouse button when the cursor is outside of the scanned image. A new window is opened which defines the entire area accessible by the X Y sensors. The maximum scanned area is defined by white dotted lines; this represents the region in which the linearize routine has selected. By pointing the cursor within the outlined green region, the scan area may be positioned anywhere within the range of the sensor. However, if the green outline is outside of the white dotted line, the scanned are is not linear and will cause a distorted image. For convenience, the Start and Stop points are indicated. When scanning at angles other than 0, 90, 180, and 270 degrees, the actual scan area is marked by the red outline.

Extra Zoom

By double-clinking the left mouse button when the cursor is within the scanned image, a new window is opened. The grid region represents the current zoomed area. The user may zoom 2x or 4x anywhere within the area by clicking on the zoom buttons at bottom of the window and moving the outlined area by pointing the cursor within the area and dragging the outline while pressing the left mouse button. Clicking the Apply button locks in the new scan region.

Force Distance Curve

This function allows the user to measure a force-distance curve at any arbitrary location within a scan area. Typically, the F-D curve refers to measuring the Error signal Z(Err), which is the cantilever deflection, as a function of the Z actuator position. It represents how the cantilever bends as the tip approaches the surface to contacts, and the degree of adhesion of the tip onto the surface on retraction. There may or may not be deformation of the surface, depending on the hardness of the surface relative to the stiffness of the cantilever.

Select the F-D location by pressing the control key and clicking the left mouse key when the cursor is pointing at the location within the scanned image display. A black dot appears on the scanned image. By selecting "Force-Distance Curve" under the pull-down Tools menu, a new window appears for selecting the F-D parameters, activating the D-D routine and displaying the approach (Curve #1, green line) and retract curves (Curve #2, red line) provided the cantilever is sufficiently stiff. The F-D curve may be obtained when the Z PID loop is in feedback and the Z actuator (Z(Hgt) is in mid-scale.

The range of the Z actuator is approximately 10 microns. The Z-DAC voltage is the inverse of the extension: The actuator is fully retracted at 10000 mV and fully extended at 0 mV. The Start position should be a higher value (less extended) than the Stop position. If Z(Hgt) is about mid-scale (5000 mV), then a typical Start position might be 8000 mV and Stop position at initially 3-4000 mV. The degree of deflection of the cantilever relative to the Z actuator position is dependent on a number of factors. The user should refer to the large body of F-D literature for further understanding of the nature and implications of the measurement.

In addition to the usual scaling parameters for the display, the user may select the resolution (Pixels, usually 256) and the rate of data acquisition. When the data rate is set to 0 ms/pixel, the data acquisition rate is set for maximum, about 15 to 25 microseconds, depending on the controller processor speed.

DISPLAY

Color palette
This feature opens a directory that contains the available color palettes that can be selected for displaying images.

WINDOW

The contents listed are the windows that have been opened in the Main Window.

HELP

- Additional information on the various functions listed above.
- Information on the version of image acquisition software that is currently in operation.

Appendix K: Customer Communication

For your convenience, this appendix contains forms to help you gather the information necessary to help us solve your technical problems and a form you can use to comment in the product documentation. When you contact us, we need the information on the Technical Support Form and the configuration form, if your manual contains one, about your system configuration to answer your questions as quickly as possible.

Pacific Nanotechnology, Inc. has technical assistance through electronic, fax, and telephone systems to quickly provide the information you need. Our electronic services include an e-mail support. If you have a hardware or software problem, first try the electronic support system. If the information available on these systems does not answer you questions, we offer fax and telephone support through our technical support center, which are staffed by application engineers.

Electronic Services

E-Mail Support

You can submit technical support questions through e-mail at the Internet address listed below. Remember to include your name, address, and phone number so we can contact you with solutions and suggestions.

Fax and Telephone Support

Pacific Nanotechnology, Inc.	**Telephone**	**Fax**
Headquarters 3350 Scott Blvd #29 Santa Clara, CA 95054-3105	(408) 982-9492	(408) 982-9151

Technical Support Form

Photocopy this form and update it each time you make changes to your software or hardware, and use the completed copy of this form as a reference for your current configuration. Completing this form accurately before contacting Pacific Nanotechnology, Inc. for technical support helps our applications engineers answer your questions more efficiently.

If you are using any Pacific Nanotechnology, Inc. hardware or software products related to this problem, include the configuration forms from their user manuals. Include additional pages if necessary.

Name _____

Company _____

Address _____

_____

Fax ( ) _____ **Phone** ( ) _____

Computer Brand _____ **Model** _____ **Processor** _____

Operating system (include version number) _____

Clock speed _____ MHZ **RAM** _____ MB **Display adapter** _____

Mouse ___ yes ___ no **Other adapters installed** _____

Hard disk capacity _____ MB **Brand** _____

Instruments used _____

_____

PNI hardware product model _____ **Revision** _____

Configuration _____

PNI software product _____ **Version** _____

Configuration _____

The problem is: _____

_____

_____

Error messages: _____

_____

The following steps reproduce the problem: _____

_____

_____

_____

Documentation Comment Form

Pacific Nanotechnology, Inc. encourages you to comment on the documentation supplied with our products. This information helps us provide quality products to meet your needs.

Title: *PScan2™ Controller User Manual*

Edition Date: *January 2002*

Part Number:

Please comment on the completeness, clarity, and organization of the manual.

If you find errors in the manual, please record the page numbers and describe the errors.

Thank you for your help.

Name

Title

Company

Address

Phone ( )

Mail to: Pacific Nanotechnology, Inc.

Headquarters
3350 Scott Blvd #29
Santa Clara, CA 95054-3105

Fax to: Pacific Nanotechnology, Inc.

FAX: (408) 982-9151